



**Maribel da Rocha Aires Um sistema de ensino assistido em cálculo com
várias variáveis**



Maribel da Rocha Aires Um sistema de ensino assistido em cálculo com várias variáveis

dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Ensino de Matemática, realizada sob a orientação científica do Dr. Luís Descalço, Professor Auxiliar do Departamento de Matemática da Universidade de Aveiro

Dedico esta dissertação ao meus pais, sem a ajuda dos quais não conseguiria ter concretizado a minha realização profissional; ao Roberto, meu namorado, pelo apoio permanente e ao meu irmão, cunhada e sobrinhas por todo o estímulo que me deram.

o júri

presidente

Prof. Dr. Helmuth Robert Malonek

Professor Catedrático da Universidade de Aveiro

Prof. Dr. José Luís Pires Ramos

Professor Associado da Universidade de Évora

Prof. Dr. Luís António Arsénio Descalço

Professor Auxiliar da Universidade de Aveiro (Orientador)

agradecimentos

Para a escrita desta dissertação contei com o apoio de várias pessoas, às quais pretendo fazer o meu agradecimento.

De uma forma particular e especial gostaria de agradecer ao meu orientador Professor Doutor Luís Descalço por todo o apoio, incentivo, dedicação, empenho e disponibilidade que sempre demonstrou durante a supervisão desta dissertação.

Ao Professor Doutor Batel Anjo e a toda a equipa do Projecto Matemática Ensino (PmatE) que me apoiaram aquando da minha colaboração neste projecto, a qual determinou a escrita desta dissertação.

E a todos aqueles que directa ou indirectamente me apoiaram e me motivaram para a escrita desta dissertação, em particular às pessoas a quem dedico este trabalho, o meu muito obrigada.

palavras-chave

autómatos, autómatos probabilísticos, ensino assistido por computador e sistema inteligente de ensino assistido por computador.

resumo

Nas últimas décadas, com a crescente presença dos computadores na vida moderna, tem-se debatido, de forma intensa, a questão da integração das novas tecnologias da informação e comunicação no processo de ensino-aprendizagem.

Neste trabalho é apresentada uma proposta de um sistema de ensino assistido por computador, que faz uso da plataforma de ensino assistido do Projecto Matemática Ensino da Universidade de Aveiro, inspirado na teoria dos autómatos probabilísticos.

A dissertação é composta por três partes. Uma primeira parte sobre matemática, concretamente sobre autómatos e autómatos probabilísticos, os quais constituem o suporte matemático do nosso sistema de ensino assistido por computador. Na segunda parte é feita referência a várias experiências de ensino assistido por computador e sistemas de ensino assistido, e em particular, é referido o Projecto Matemática Ensino da Universidade de Aveiro. Finalmente, na terceira parte, apresentamos a nossa proposta para um sistema inteligente de ensino assistido por computador, algumas reflexões e propostas de trabalho futuro.

keywords

automata, probabilistic automata, computer-aided instruction and intelligent tutoring systems

abstract

In the last decades, with the increasing presence of computers in modern live, the integration of new information and communication technologies in education has been a matter of intense debate.

In this work, we propose an intelligent computer assisted tutoring system, which uses the software learning platform of Projecto Matemática Ensino from the University of Aveiro, inspired by the theory of probabilistic automata.

The thesis has three parts. The first part is about mathematics, namely, about automata and probabilistic automata, which constitute the mathematics supporting our computer system. In the second part, we refer several studies about computer aided learning and computer learning systems, and in particular, we refer the Projecto Matemática Ensino, from the University of Aveiro. Finally, in the third part, we propose our intelligent computer aided learning system, and we end up with some final remarks and propose some future work.

Um sistema de ensino assistido em cálculo com
várias variáveis

Maribel da Rocha Aires

Universidade de Aveiro

14 de Fevereiro de 2007

Conteúdo

1	Introdução	1
1	Estrutura	1
2	Autómatos	2
3	Computadores no ensino	5
4	Sistemas inteligentes	7
2	Autómatos	9
1	Autómatos finitos	11
1.1	Autómatos e Linguagens Formais	14
1.2	Autómatos determinísticos	14
1.3	Autómatos não-determinísticos	15
1.4	Autómatos determinísticos e não-determinísticos	17
1.5	Operações com autómatos finitos	20
2	Autómatos Probabilísticos	27
2.1	Linguagem aceite por um autómato probabilístico	30
2.2	Autómatos Probabilísticos e Redes Neurais	30
2.3	Autómatos Probabilísticos e Modelos Escondidos de Markov	33
3	Aplicações	35
3	Ensino Assistido por computador	39
1	O computador ao longo dos tempos	40
2	O computador na sociedade	42
3	O computador na escola	45
4	Introdução das novas tecnologias nas escolas de Portugal	46
4.1	Projecto Minerva	46
4.2	Edutic	48

4.3	Programa Nónio - séc XXI	48
4.4	CRIE (Computadores, Redes e Internet na Escola)	48
5	Introdução das novas tecnologias nas escolas do mundo	49
6	Ensino assistido por computador(EAC)	53
6.1	Modos de Ensino Assistido por Computador	53
6.2	Porquê o Ensino Assistido por Computador	55
4	Projecto Matemática Ensino (PmatE)	59
1	O projecto	59
2	Modelos Geradores de Questões (MGQ)	60
3	A Experiência de Cálculo III	61
5	Sistema de Ensino Assistido por Computador	67
1	Sistemas Inteligentes	68
2	A motivação	72
3	Armanezamento de Informação	72
4	Funcionamento do Sistema	73
5	Reflexões	76
A	Código C#	83
B	Tabelas	101
C	Modelos	105

Capítulo 1

Introdução

“A cultura matemática de hoje será válida também para o Homem de amanhã se for dada numa forma construtiva.”

Emma Castelnuovo

Tendo presente a colaboração no Projecto Matemática Ensino da Universidade de Aveiro, no âmbito da disciplina de Cálculo III, decidimos rentabilizar a plataforma de ensino assistido que este projecto já possui e que é utilizada como ferramenta de ensino e de avaliação em várias disciplinas desta universidade.

O uso de computadores no ensino tem sido alvo de grandes investigações nos últimos anos. Acreditamos que o uso das novas tecnologias, em particular o computador, constitui uma mais valia para o ensino, aprendizagem e avaliação em matemática.

Com este trabalho, propomo-nos a apresentar a nossa proposta para um sistema de ensino assistido por computador inovador, para o cálculo com várias variáveis, que faz uso da plataforma de ensino assistido do Projecto Matemática Ensino da Universidade de Aveiro.

1 Estrutura

A presente dissertação encontra-se dividida em três partes. A primeira sobre matemática, concretamente sobre autómatos e autómatos probabilísticos, os quais constituem o suporte matemático do nosso sistema de ensino assistido

por computador. Na segunda parte é feita referência a várias experiências de ensino assistido por computador e sistemas de ensino assistido, e em particular, é referido o Projecto Matemática Ensino da Universidade de Aveiro. Finalmente, na terceira parte, apresentamos a nossa proposta para um sistema inteligente de ensino assistido por computador e algumas reflexões que constituem propostas de trabalho futuro. Estas três áreas temáticas encontram-se divididas em quatro capítulos, para além de um capítulo introdutório. No primeiro capítulo é feita a introdução ao trabalho, onde se dá a conhecer, de forma resumida, o conteúdo da dissertação. No segundo capítulo abordamos os autómatos a nível da sua definição, propriedades e aplicações, dando especial ênfase aos autómatos probabilísticos. Fazemos referência à ligação existente entre autómatos probabilísticos e redes neuronais e ainda entre autómatos probabilísticos e modelos escondidos de Markov. Apresentamos ainda algumas aplicações dos autómatos a áreas, como por exemplo, a biologia. O terceiro capítulo trata do ensino assistido por computador numa perspectiva histórica quer em Portugal, quer no estrangeiro. Iniciamos o capítulo com uma breve resenha histórica do computador e a sua introdução na sociedade, seguindo-se a apresentação de alguns estudos que foram realizados nesta área, assim como medidas implementadas para a introdução dos computadores nas escolas, portuguesas e estrangeiras. O quarto capítulo diz respeito ao caso particular do Projecto Matemática Ensino, em funcionamento na Universidade de Aveiro, o qual prestou o seu apoio no desenvolvimento deste trabalho. Destinamos a este projecto o capítulo 4 no qual referimos o material desenvolvido no âmbito do projecto e analisamos a experiência decorrida com a disciplina de Cálculo III da Universidade de Aveiro, durante o primeiro semestre do ano lectivo 2005/2006. Para finalizar, no capítulo 5, falamos do modo de funcionamento dos sistemas inteligentes de ensino assistido por computador e apresentamos a nossa proposta. Terminamos com algumas reflexões que podem estimular novos trabalhos nesta área.

2 Autómatos

O interesse em encontrar formas de fugir aos trabalhos rotineiros e repetitivos deu origem à construção, em grande escala, de máquinas, por volta do

século XVIII. Ocorreu nesta altura a revolução industrial. No entanto, com o desenvolvimento tecnológico, mais visível a partir da segunda guerra mundial, foram construídas máquinas cada vez mais eficientes e que substituem o Homem, essencialmente no que concerne às tarefas de processamento de informação, dando origem à denominada revolução da informação.

O interesse pelo estudo das propriedades inerentes aos sistemas de informação utilizados deu origem a uma nova área científica, chamada *teoria dos autómatos*. O seu objectivo primeiro é descrever o comportamento dos sistemas de reconhecimento de padrões e de controle. O comportamento deste tipo de sistemas é determinado pelo modo como este é construído e pelo seu modo de funcionamento interno. Para determinar o sistema é preciso conhecer as entradas do sistema, o conjunto de estados e o conjunto de estados de saída. É igualmente importante conhecer o modo como estes conjuntos se relacionam, a que chamamos *transição do sistema*. Uma das formas mais habituais de representar um autómato é através de um grafo. Os nós do grafo, representados por círculos, correspondem aos estados do autómato; as entradas estão simbolizadas por meio de setas, etiquetadas pelo símbolo de entrada, e ligam um estado a outro estado do autómato. Nos autómatos surge ainda a noção de estado inicial o qual corresponde ao estado em que se encontra o autómato antes de qualquer entrada.

Um exemplo de um autómato simples é uma máquina de café que aceita moedas de 5 cêntimos, 10 cêntimos e 20 cêntimos e devolve um café por 25 cêntimos. As moedas são as entradas da máquina, e a máquina aceita e reconhece qualquer sequência de moedas que perfaça um total de 25 cêntimos. O funcionamento da máquina de café encontra-se ilustrado na Figura 1.1.

Como se pode verificar, cada sequência de entrada corresponde a um percurso entre o estado inicial, q_0 , e o estado final, q_{25} .

De um modo geral, podemos afirmar que um autómato finito é formado por:

- um conjunto de entradas;
- um conjunto de estados, incluindo o estado inicial e os estados finais;
- uma função de transição, responsável pelo comportamento do autómato que a cada par (entrada, estado) faz corresponder o estado de saída.

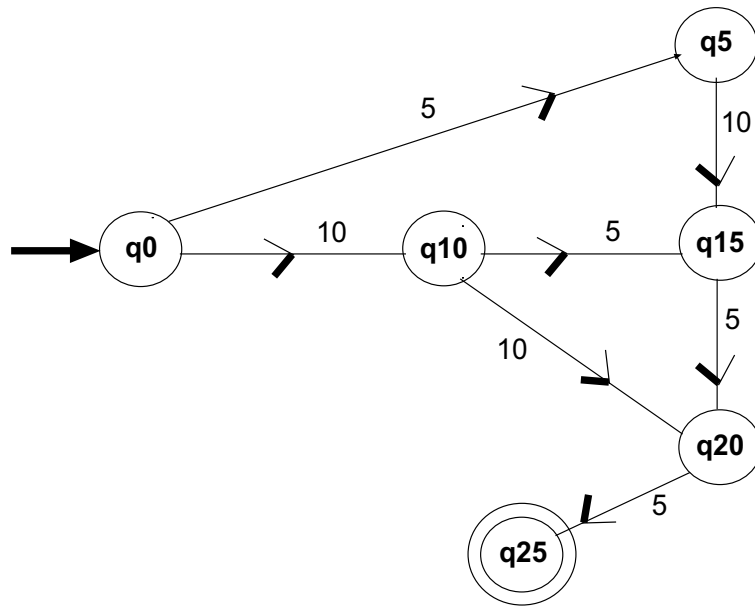


Figura 1.1: Máquina de Café

Os autômatos probabilísticos são uma generalização dos autômatos em que um símbolo de entrada pode levar o autômato a qualquer estado com uma certa probabilidade. Pensemos, por exemplo, no lançamento de um dado equilibrado. O conjunto de estados do autômato é o conjunto $Q = \{1, 2, 3, 4, 5, 6\}$; as entradas do autômato são os lançamentos do dado; a função de transição é a aplicação que a cada estado, atendendo ao lançamento ocorrido, faz corresponder o estado de saída. Note-se que, como se trata de uma dado equilibrado, a cada transição está associada a probabilidade de $\frac{1}{6}$, ou seja, dado um estado e um lançamento, cada um dos estados do autômato poder ser o próximo estado de saída com a probabilidade de um $\frac{1}{6}$. Se quisermos modelar um dado não equilibrado basta associar a cada transição a respectiva probabilidade.

De um modo geral, podemos afirmar que um autômato probabilístico finito é formado por:

- um conjunto de entradas;
- um conjunto de estados, incluindo o estado inicial e os estados finais;
- uma função de transição, responsável pelo comportamento do autômato que, a cada par (entrada, estado) faz corresponder o estado de saída;

- uma matriz de probabilidades $[p_{ij}]$, onde para cada entrada, a , p_{ij} é a probabilidade do autómato transitar do estado q_i para o estado q_j sob a entrada a .

3 Computadores no ensino

A partir do ano de 1982, considerado pela revista *Time*, como o ano do computador dado o forte contributo dos computadores pessoais no trabalho humano, assistiu-se a um crescente desenvolvimento e aperfeiçoamento desta máquina.

O computador entrou na sociedade e rapidamente invadiu a vida pessoal de grande parte da população mundial. Com a exigência da sociedade actual em constante mutação e com a presença marcante das novas tecnologias, surgiu a necessidade de repensar o processo de ensino-aprendizagem, numa perspectiva de alteração dos métodos tradicionais de ensino. Em vez de obrigar o aluno a reter grandes quantidades de informação, é urgente levar o aluno a compreender e aplicar os seus conhecimentos a situações novas. O ensino dos dias de hoje tem que ensinar o aluno a pensar e motivá-lo para uma aprendizagem ao longo da vida, passando o aluno a ter um papel activo no processo de ensino aprendizagem. No início, os computadores eram vistos como uma ameaça aos docentes. Muitos eram os professores que receavam ser substituídos pelos computadores. Com o passar do tempo, esse receio foi-se desvanecendo e, tal como afirma Ponte, “Será a combinação dos dois (professor e computador), ambos no máximo das suas potencialidades, que constituirá a equipa pedagógica do futuro.” [8]

Foram inúmeras as medidas tomadas para a integração do computador nas escolas portuguesas e estrangeiras, sendo algumas apresentadas no capítulo 3.

Um dos projectos que mais tem contribuído para o uso do computador, no âmbito da disciplina de matemática, nas salas de aula das escolas portuguesas é o Projecto Matemática Ensino da Universidade de Aveiro. Este projecto tem desenvolvido desde 1990 uma plataforma de ensino assistido (PeA). Surgiu, e segundo palavras do fundador, o Professor Doutor David Vieira do Departamento de Matemática da Universidade de Aveiro, de “um desafio a dois colegas mais novos - António Batel Anjo e Maria Paula Carvalho - para criarmos um

sistema informático de avaliação em disciplinas do ensino superior com grande número de alunos”. Devido à insuficiência de meios naquele tempo, decidiram experimentar o seu sistema informático em alunos dos 7º ano de escolaridade de uma escola de Aveiro. Atendendo ao sucesso que obtiveram, a equipa sentiu-se encorajada para continuar a desenvolver cada vez mais o seu sistema informático e ampliar o sistema de diversos modos. A nível de competições contam já com o EQUA_{mat} (3º Ciclo), mat12 (secundário), MINImat (1º ciclo) e MAIS_{mat} (2º ciclo). É de realçar ainda a contribuição da plataforma do PmatE a nível do Ensino Superior. Nos últimos anos esta plataforma tem sido utilizada para efeitos de avaliação em algumas disciplinas da Universidade de Aveiro, nomeadamente, Cálculo I, Cálculo II, Cálculo III, Sistemas Digitais, entre outras. Actualmente encontra-se apenas disponível na Internet, abrangendo, como já foi referido, vários graus de ensino, desde o básico até ao superior. Fazemos notar ainda a existência de outras iniciativas do projecto com o intento de promover o gosto pela matemática junto dos mais novos e apoiar o ensino desta disciplina em Portugal e no estrangeiro.

O trabalho essencial do PmatE é o desenvolvimento de modelos de software, chamados Modelos Geradores de Questões (MGQ), essencialmente para matemática. Cada MGQ é constituído por um texto principal e por, pelo menos, quatro questões de resposta verdadeiro/falso, podendo conter ou não imagens. Após a criação do modelo este é classificado, quer do ponto de vista científico-pedagógico, quer do ponto de vista da sua dificuldade. Com efeito, cada MGQ é etiquetado por um chamado *objectivo secundário* que corresponde ao tema que se encontra abordado no MGQ. Os conteúdos e respectivos objectivos de todas as áreas e graus de ensino abrangidos pelo PmatE encontram-se organizados na denominada *árvore de objectivos*. Uma das maiores vantagens destes MGQ é o facto de um único modelo permitir gerar um grande número de questões. A título de exemplo, num modelo, em vez de termos “ $1+1=2$ ”, temos “ $a + b = c$ ”, onde $a, b \in \{1, 2\}$ e $c \in \{0, 1, 2\}$. Assim, concretizações possíveis para este modelo seriam: $1 + 0 = 2$ ou $0 + 1 = 1$. Exemplos mais elaborados são apresentados no capítulo 4.

4 Sistemas inteligentes

Como vimos na secção anterior, a plataforma de ensino assistido do PmatE fornece aos alunos, em cada concretização do MGQ, um conjunto de questões pré-determinadas cuja escolha é independente do conhecimento efectivo de cada aluno. O professor, ao criar uma prova nesta plataforma, está a incluir questões sobre um ou mais temas, que vão ser iguais, a menos da aleatoriedade imposta pelo próprios MGQ, para todos os alunos a quem ele aplicar a prova. Este tipo de sistemas reage do mesmo modo com todos os seus utilizadores e de maneira semelhante à seguinte: se a resposta 21 estiver correcta, então o aluno deve responder à questão 54, caso contrário, responde à questão 32. Neste tipo de sistemas, o aluno e as suas capacidades e dificuldades individuais são esquecidos. Dada a diversidade de alunos existente nas escolas, sentiu-se a necessidade de se criar um novo tipo de sistemas de ensino assistido por computador que, em oposição aos anteriores, “olhassem” para o aluno como um ser singular: *os sistemas de ensino assistido por computador inteligentes*. Estes novos sistemas de ensino assistido por computador reagem de acordo com o desempenho de cada aluno e são compostos por cinco camadas: conhecimento, aluno, pedagógica, comunicação e “expert”. A camada do conhecimento é preenchida com os conteúdos temáticos que vão ser apresentados aos alunos. A camada do aluno é responsável por reter o conhecimento do aluno, com base no seu desempenho. Na camada pedagógica, o sistema, atendendo à performance do aluno e ao seu conhecimento, decide quais as novas questões a apresentar ao aluno. A camada da comunicação é a parte do sistema que interage directamente com o aluno. Por fim, a camada “expert” é uma espécie de aluno virtual inteligente com o qual o sistema compara as respostas dadas pelo aluno real.

Inspirados por este novo tipo de sistemas, decidimos rentabilizar a plataforma de ensino assistido por computador do PmatE, construindo uma proposta de um sistema inteligente, tomando para ensaio os conteúdos programáticos do cálculo com várias variáveis.

Actualmente, o nosso sistema é formado por 41 estados que correspondem, cada um, a um tema ou sub-tema dos conteúdos programáticos do cálculo com várias variáveis. A transição entre conteúdos é feita com uma certa proba-

bilidade que é sucessivamente actualizada ao longo da execução do programa. Toda a informação necessária para a criação deste sistema de ensino assistido encontra-se arrecadada em quatro matrizes A, B, C e D, respectivamente, permanência, bases, conhecimento e distância. A matriz B representa a dependência entre os diferentes temas e sub-temas. Esta matriz fornece a informação sobre quais os conteúdos que são base para outros. A matriz D guarda a informação sobre a proximidade dos diversos temas. Quanto maior o valor do elemento da matriz, mais próximos estão os temas. Estas duas matrizes contêm informação que não se altera ao longo da execução do programa. No que diz respeito à matriz C esta é, na realidade, um vector com quarenta e uma posições que conserva a percentagem de conhecimento que o aluno possui em cada um dos objectivos secundários. Finalmente, a matriz A também é, na verdade, um vector com quarenta e uma posições, que contém o número de questões respondidas pelo aluno por cada tema ou sub-tema. Os vectores A e C guardam informação que é actualizada após cada interacção do aluno. Inicialmente, todas as posições do vector C tomam o valor de 50 por cento e todas as posições do vector A são nulas. Tendo em conta estes valores e os valores constantes das matrizes B e D, o sistema cria o vector de probabilidades, P, utilizando uma fórmula que será apresentada no capítulo 5. Com os valores de P o primeiro estado é escolhido e um MGQ sobre esse tema ou sub-tema é aleatoriamente seleccionado e apresentado ao aluno. Realçamos que os valores iniciais do vector P são tais que o primeiro estado a ser escolhido não corresponde a um tema com elevado grau de dificuldade. Após o aluno ter respondido ao modelo que lhe foi apresentado, validando as quatro afirmações, como verdadeiras ou falsas, o sistema guarda o número de respostas certas dadas pelo aluno. Actualizados os vectores A e C, um novo vector de probabilidades é criado, um novo estado é seleccionado e um modelo é aleatoriamente escolhido, repetindo-se todo o processo anteriormente descrito até que o aluno demonstre ter algum conhecimento nos temas abordados.

Capítulo 2

Autómatos

“Os animais são meras máquinas vivas ou autómatos como os relógios, capazes de comportamento complexo, mas desprovidos de alma ou mentes.”

Descartes

“somos(humanos)autómatos conscientes.”

Huxley

O Homem está continuamente à procura de meios e máquinas que o substituam nos trabalhos rotineiros.

Por volta dos meados do século XVIII o Homem começou a desenvolver, a grande escala, máquinas que lhe facilitavam ou o substituíam nos trabalhos físicos sucedendo-se, assim, a revolução industrial.

A partir da segunda guerra mundial e até aos nossos dias é bem visível o rápido desenvolvimento da ciência e da tecnologia, essencialmente no que diz respeito aos computadores, telemóveis, entre outros. Podemos dizer que estamos perante uma revolução a nível da informação. Têm-se desenvolvido máquinas que substituem o Homem, essencialmente no que concerne às tarefas de processamento de informação. No entanto, ao contrário do que se poderia esperar, o Homem continua a trabalhar muitas horas por dia. Tal facto pode, porventura, ser explicado pelo acréscimo da complexidade das máquinas utilizadas e pelos crescentes desafios diários que são colocados ao ser humano.

O interesse pelo estudo das propriedades inerentes aos sistemas de informação utilizados deu origem a uma nova área científica, denominada por teo-

ria de autómatos. Basicamente, a teoria de autómatos pode ser descrita como sendo o estudo da dinâmica de sistemas de busca de padrões.

No entanto, a ideia de autómato surgiu já no século XVII impulsionada pelo determinismo científico da época. O crescente conhecimento do mundo físico fez com que se comparasse o universo a uma máquina, tal como Descartes (1596 - 1650) já tinha especulado acerca dos animais e até mesmo do ser humano.

Uma das mais importantes máquinas teóricas é a Máquina de Turing, em honra a Alan Turing que em 1930, mesmo antes de existirem os computadores, desenvolveu uma máquina teórica abstracta que continha as mesmas características que um computador actual. A máquina de Turing descreve o que um computador é ou não capaz de fazer. As conclusões a que Turing chegou com o estudo da sua máquina continuam válidas para as máquinas dos dias de hoje, [5] e [20].

A teoria de autómatos não é uma área da matemática de aplicação exclusiva às ciências da computação. Sabemos da ligação desta área de investigação com o processamento de informação ao nível do sistema nervoso humano, à análise de sistemas de informação - transmissão, à reacção do Homem perante o seu meio ambiente, à teoria de jogos, entre outras aplicações. Exemplos concretos serão apresentados na secção 3.

No entanto, existem sistemas cuja transição é descrita de modo probabilístico. Para a análise e explicação deste tipo de sistemas recorreremos aos *autómatos probabilísticos*. Os autómatos probabilísticos começaram a aparecer na literatura sob diferentes pontos de vista. Shannon e Weaver, em 1948, iniciaram o estudo dos autómatos probabilísticos motivados pela teoria da informação. Mais tarde, em 1958, Ashby apresentou uma definição, ainda que vaga, de autómato probabilístico. Porém, o verdadeiro desenvolvimento da teoria dos autómatos probabilísticos aconteceu a partir dos anos sessenta quando cientistas das diferentes partes do mundo introduziram os autómatos probabilísticos como uma generalização dos autómatos determinísticos, [29]. Os diferentes tipos de autómatos, assim como as suas diversas designações, estão relacionadas com o seu comportamento. Neste capítulo dedicamo-nos ao estudo dos autómatos determinísticos, não-determinísticos e probabilísticos na sua forma finita, ou seja, autómatos em que todos os seus constituintes são finitos. Portanto, sempre que aparecer a palavra autómato no decorrer do

texto esta significará invariavelmente autómato finito.

1 Autómatos finitos

Como já referimos, o objectivo primeiro da teoria de autómatos é descrever a dinâmica de sistemas de controle e reconhecimento de padrões. Para determinar o sistema é necessário conhecer as entradas do sistema (uma sequência de símbolos de um conjunto finito, Σ), o conjunto finito de estados, Q , e o conjunto finito de estados de saída, F . É igualmente importante conhecer o modo como estes conjuntos se relacionam, a que chamamos *transição do sistema*. O comportamento de um autómato fica explicitado pela denominada *função de transição*, δ , definida em $Q \times \Sigma$. Uma das formas mais habituais de representar um autómato é através de um grafo. Os nós do grafo, representados por círculos, correspondem aos estados do autómato; as entradas estão simbolizadas por meio de setas, etiquetadas pelo símbolo de entrada, e ligam um estado a outro estado do autómato. Acontece, por vezes, que após um input o autómato permanece no mesmo estado, isto é, o estado seguinte é o estado actual. Estas situações estão retratadas nas Figuras 2.1 e 2.2.

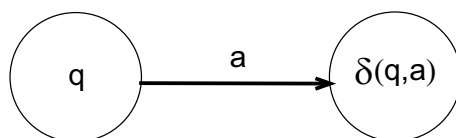


Figura 2.1: Transição para um novo estado

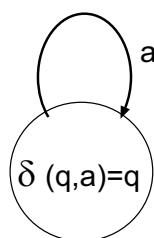


Figura 2.2: Transição para o mesmo estado

Nos autómatos surge ainda a noção de estado inicial o qual corresponde ao estado em que se encontra o autómato antes de qualquer entrada. Para

distinguir no grafo, ou diagrama de estados, o estado inicial e os estados finais dos restantes estados utilizamos um círculo com uma seta a apontar para ele para o caso do estado inicial e círculos duplos para os estados finais, como na Figura 2.3.

Um dos mais simples exemplos de autómato finito é o caso de um interruptor que memoriza se está no estado “ligado” ou “desligado”, e permite que o utilizador pressione um único botão cujo efeito será diferente de acordo com o estado em que se encontra o interruptor, ou seja, se ele estiver no estado ligado passa ao estado desligado e vice-versa. A figura 2.4 ilustra o funcionamento do interruptor.

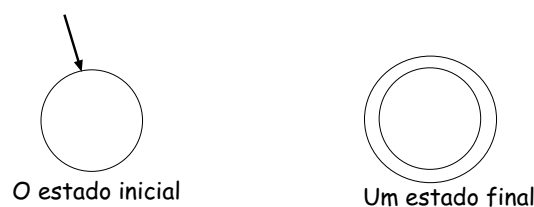


Figura 2.3: Estados inicial e final

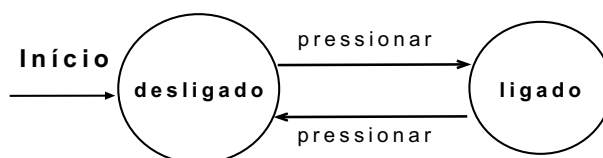


Figura 2.4: Interruptor

Os estados estão representados por círculos e a “entrada” está representada por setas. Neste caso temos apenas dois estados, o ligado e o desligado e a única acção é pressionar o botão. O estado inicial, aquele no qual se encontra o interruptor antes de qualquer acção, está indicado pela palavra ‘início’ e por uma seta.

Um exemplo um pouco mais elaborado de autómato finito é uma máquina de café que aceita moedas de 5 cêntimos, 10 cêntimos e 20 cêntimos e devolve um café por 25 cêntimos. As moedas são as entradas da máquina, e a máquina aceita e reconhece qualquer sequência de moedas que perfaça um total de 25 cêntimos. A “linguagem” aceite pela máquina de café é o seguinte conjunto de

sequências de entrada

$$\{(20, 5); (5, 20); (10, 10, 5); (10, 5, 10); (5, 10, 10); (5, 5, 5, 5, 5)\},$$

onde linguagem significa o conjunto de todas as sequências de zero ou mais símbolos justapostos que se podem formar sobre um determinado alfabeto. Este conceito é fundamental na teoria da computação e também muito utilizado na teoria de autómatos e para o qual reservamos a secção seguinte.

A máquina de café tem seis estados $q_0, q_5, q_{10}, q_{15}, q_{20}, q_{25}$, (o índice de cada estado equivale ao montante recebido pela máquina). O estado q_0 , corresponde ao instante em que o utilizador começa a introduzir as moedas, é o chamado *estado inicial*; o estado q_{25} , representa o momento em que a máquina devolve o café ao utilizador, é o chamado *estado final*. O funcionamento da máquina de café encontra-se ilustrado na Figura 2.5.

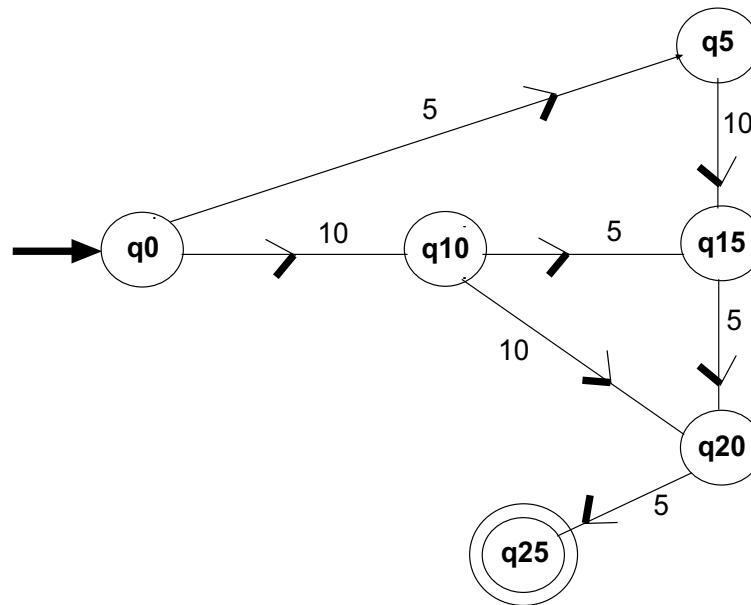


Figura 2.5: Máquina de Café

Como se pode verificar, cada sequência de entrada corresponde a um percurso entre o estado inicial, q_0 , e o estado final, q_{25} .

Para mais pormenores, consultar [18] e [20].

1.1 Autómatos e Linguagens Formais

Antes de definirmos linguagem sobre um alfabeto, vejamos o que se entende por alfabeto. Um *alfabeto*, Σ , é um conjunto finito de símbolos ou caracteres. Dizemos que $w = \sigma_1\sigma_2\ldots\sigma_n$ é uma *palavra* de Σ , onde $\sigma_i \in \Sigma$ e $i = 1 \ldots n$. Chamamos *comprimento* de uma palavra ao número de elementos do alfabeto que a palavra possui. É usual considerar a *palavra vazia*, denotada por ϵ , a qual tem comprimento 0. São ainda utilizadas a notação Σ^+ que representa o conjunto de todas as palavras possíveis em Σ e $\Sigma^* = \Sigma^+ \cup \{\epsilon\}$. Uma *linguagem formal* ou simplesmente *linguagem* é um conjunto de palavras sobre um alfabeto. Diz-se que uma linguagem é reconhecível se for aceite por um autómato. A classe das linguagens reconhecíveis coincide com a classe das linguagens racionais. Estas são todas as linguagens que se podem obter a partir de conjuntos finitos usando as operações seguintes:

- (i). $A \cup B = \{w : w \in A \text{ ou } w \in B\}$;
- (ii). $A \cdot B = \{wv : w \in A, v \in B\}$;
- (iii). $A^+ = \{w_1w_2\ldots w_n : w \in A, n \geq 1\}$;
- (iv). $A^* = A^+ \cup \{\epsilon\}$.

1.2 Autómatos determinísticos

Um *autómato finito determinístico* é um autómato que se encontra sempre em um único estado após ler uma qualquer sequência de entrada. A designação “determinístico” implica que existe um e somente um estado para o qual o autómato pode transitar a partir do seu estado actual. A definição formal de autómato finito determinístico é dada a seguir:

Definição 2.1. [Autómato finito determinístico]

Um *autómato finito determinístico* é um quintuplo $\langle Q, \Sigma, \delta, q_0, F \rangle$, tal que:

- (i). Q é um conjunto finito de estados;
- (ii). Σ é um conjunto finito de símbolos, ao qual chamamos *alfabeto*;

- (iii). $\delta : Q \times \Sigma \rightarrow Q$ é a *função de transição*;
- (iv). q_0 é o *estado inicial*, isto é, o estado em que se encontra o autómato antes de qualquer entrada. É claro que $q_0 \in Q$;
- (v). F é um subconjunto dos estados de Q , isto é, $F \subseteq Q$, cujos elementos são chamados *estados finais* do autómato.

Pelo facto de se tratar de um autómato determinístico, a função de transição, δ , determina univocamente o próximo estado do autómato. Assim, se o autómato se encontra no estado $q \in Q$ e lê a entrada $a \in \Sigma$, o autómato irá transitar para o estado $\delta(q, a)$. De facto, se considerarmos entradas sucessivas de símbolos de Σ em $q \in Q$, podemos estender, recursivamente, a acção δ a Σ^* da maneira seguinte:

$$\begin{aligned}\delta(q, \epsilon) &= q \quad (q \in Q) \\ \delta(q, wa) &= \delta(\delta(q, w), a) \quad (q \in Q, w \in \Sigma^*, a \in \Sigma)\end{aligned}$$

Por vezes, e para simplificação da notação, escreve-se qw em vez de $\delta(q, w)$. Esta forma de escrita simplificada é usada no exemplo 2.2.

Dizemos que um autómato determinístico, \mathcal{M} , *reconhece* ou *aceita* uma palavra $w \in \Sigma^+$ se $\delta(q_0, w) \in F$. A *linguagem*, $L(\mathcal{M})$, *reconhecida* pelo autómato, \mathcal{M} , é o conjunto de todas as palavras aceites pelo autómato.

Exemplo 2.2. Consideremos o autómato, \mathcal{M} , em que $Q = \{0, 1, 2, 3\}$; $\Sigma = \{a, b\}$; $q_0 = 1$ e $F = \{1, 2, 3\}$, cujo diagrama de estados se encontra na Figura 2.6.

É fácil ver que as palavras $b^2a^3b^2a^4$ e b^3a^4b são reconhecidas pelo autómato uma vez que $1(b^2a^3b^2a^4) = 2$ e $1(b^3a^4b) = 3$. Pelo contrário, a palavra $b^2a^3ba^4$ não é reconhecida por \mathcal{M} dado que $1(b^2a^3ba^4) = 0$. Observa-se facilmente que $L(\mathcal{M}) = (b^*a^+b^2) \cdot (b^* \cup b^*a^+ \cup b^*a^+b)$.

1.3 Autómatos não-determinísticos

Em contraste com o que acontece com os autómatos determinísticos, num *autómato finito não-determinístico*, dado um estado e uma entrada, este pode transitar para diversos estados. Formalmente:

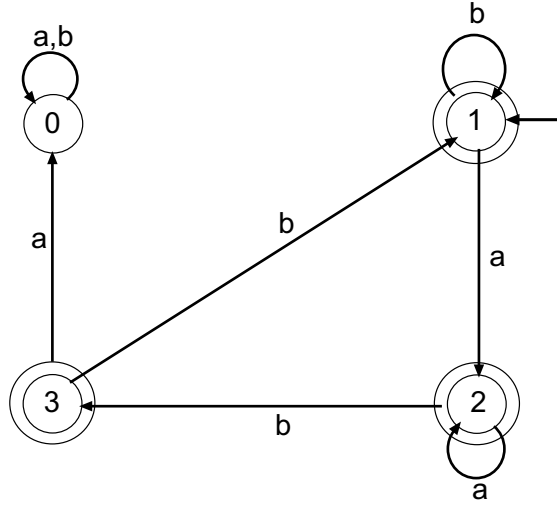


Figura 2.6:

Definição 2.3. [Autômato finito não-determinístico]

Um *autômato finito não-determinístico* é um quintuplo $\langle Q, \Sigma, \delta, q_0, F \rangle$, tal que:

- (i). Q é um conjunto finito de estados;
- (ii). Σ é um conjunto finito de símbolos, ao qual chamamos *alfabeto*;
- (iii). $\delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow \mathbb{P}(Q)$ é a *função de transição*;
- (iv). q_0 é o estado inicial, isto é, o estado em que se encontra o autômato antes de qualquer entrada. É claro que $q_0 \in Q$;
- (v). F é um subconjunto dos estados de Q , isto é, $F \subseteq Q$, cujos elementos são chamados *estados finais* do autômato.

Em oposição ao que acontece com os autômatos determinísticos estudados em 1.2, a função de transição de um autômato não-determinístico não determina exactamente qual é o próximo estado. Em vez disso, fornece um conjunto vazio ou com um número positivo de elementos para os quais a transição pode ocorrer. Esta particularidade da função de transição está especificada pela notação $\mathbb{P}(Q)$ que denota o conjunto das “partes” de Q , isto é, o conjunto de todos os subconjuntos de Q .

Pela definição, δ , aceita como argumentos $q \in Q$ e $a \in (\Sigma \cup \{\epsilon\})$ e devolve um subconjunto de Q . Formalmente, podemos definir recursivamente δ da maneira seguinte:

$$\begin{aligned}\delta(q, \epsilon) &= \{q\} \quad (q \in Q) \\ \delta(q, wa) &= \bigcup_{p \in \delta(q, w)} \delta(p, a) \quad (q \in Q, w \in \Sigma^*, a \in \Sigma)\end{aligned}$$

Do mesmo modo, e para simplificação da escrita, escreve-se qw em vez de $\delta(q, w)$. Por exemplo, se tivermos $\delta(q, a) = \{p_1, p_2, \dots, p_k\}$, entendemos que o autómato, a partir do estado actual, q , pode escolher um dos estados p_1, p_2, \dots, p_k para ser o próximo estado. No caso de $a = \epsilon$, o autómato transita para qualquer um dos seus estados. Se tivermos $\delta(q, a) = \emptyset$, não há transições possíveis a partir de q com o símbolo a .

Exemplo 2.4. Consideremos o conjunto de estados $Q = \{1, 2, 3\}$ e o alfabeto $\Sigma = \{a, b\}$. Podemos definir, por meio de uma tabela, o comportamento deste autómato como se segue:

	a	b
1	$\{1, 2\}$	$\{2, 3\}$
2	1	\emptyset
3	3	$\{2, 3\}$

Designando 1 como o estado inicial do autómato e $\{2\}$ como o conjunto dos estados finais, construímos o correspondente diagrama de estados que se encontra na Figura 2.7.

Dizemos que um autómato não determinístico, \mathcal{M} , *reconhece* uma dada palavra $w \in (\Sigma \cup \{\epsilon\})$ se $\delta(q_0, w) \cap F \neq \emptyset$. O conjunto de todas as palavras reconhecidas pelo autómato não-determinístico denomina-se por *linguagem reconhecida* pelo autómato e denota-se por $L(\mathcal{M})$.

1.4 Autómatos determinísticos e não-determinísticos

Definição 2.5. [Equivalência de Autómatos]

Sejam \mathcal{M}_1 e \mathcal{M}_2 dois quaisquer autómatos. Dizemos que \mathcal{M}_1 é *equivalente* a \mathcal{M}_2 e denotamos por $\mathcal{M}_1 \cong \mathcal{M}_2$ se e somente se $L(\mathcal{M}_1) = L(\mathcal{M}_2)$.

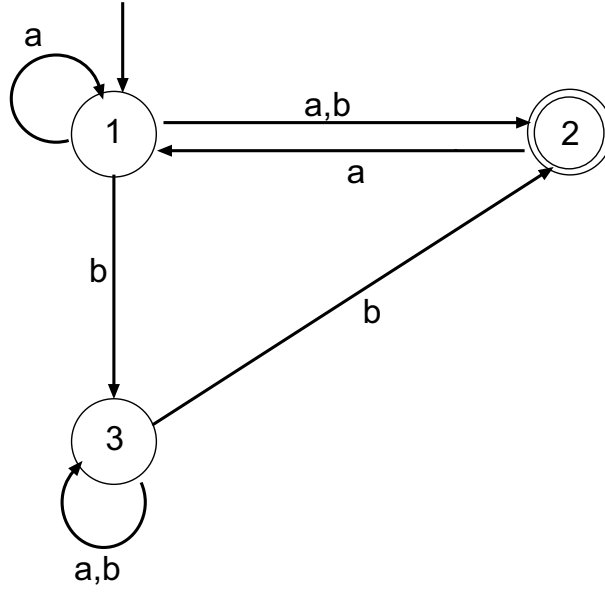


Figura 2.7:

Prova-se facilmente que esta relação é, de facto, uma relação de equivalência na classe dos autômatos, pois está definida à custa da relação de igualdade entre linguagens.

Teorema 2.6. *Seja $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F \rangle$ um autômato que reconhece a linguagem $L(\mathcal{M})$. Então existe um autômato determinístico, \mathcal{M}' , tal que $L(\mathcal{M}') = L(\mathcal{M})$.*

DEMONSTRAÇÃO

Defina-se \mathcal{M}' como um autômato com o mesmo alfabeto, Σ , de \mathcal{M} e cujo conjunto de estados é o conjunto de todos os subconjuntos de Q , $\mathbb{P}(Q)$. A função de transição $\psi : \mathbb{P}(Q) \times \Sigma \rightarrow \mathbb{P}(Q)$ é naturalmente definida como se segue:

$$\psi(P, a) = \bigcup_{p \in P} \delta(p, a) \quad (P \in \mathbb{P}(Q), a \in \Sigma).$$

O estado inicial de \mathcal{M}' é $\{q_0\}$ e os estados finais são os elementos do conjunto $F' = \{P \in \mathbb{P}(Q) : P \cap F \neq \emptyset\}$. Portanto, $\mathcal{M}' = \langle \mathbb{P}(Q), \Sigma, \psi, \{q_0\}, F' \rangle$ é um autômato determinístico. Para finalizar a demonstração falta apenas mostrar que $L(\mathcal{M}') = L(\mathcal{M})$.

$$\begin{aligned}
w \in L(\mathcal{M}) &\Leftrightarrow \delta(q_0, w) \cap F \neq \emptyset \\
&\Leftrightarrow \psi(\{q_0\}, w) \cap F \neq \emptyset \\
&\Leftrightarrow \psi(\{q_0\}, w) \in F' \\
&\Leftrightarrow w \in L(\mathcal{M}').
\end{aligned}$$

■

Atendendo ao teorema 2.6 e à definição 2.5 podemos afirmar que todo o autómato é equivalente a um autómato determinístico no mesmo alfabeto.

Exemplo 2.7. Retomamos o autómato do exemplo 2.4 cujo diagrama de estados se encontra na Figura 2.7. Vamos construir um autómato determinístico equivalente ao autómato dado seguindo os passos da demonstração do teorema 2.6. De acordo com a demonstração do teorema os estados do novo autómato são todos os subconjuntos do conjunto dos estados do autómato de base. Assim o conjunto dos estados do novo autómato será constituído pelos $2^3 = 8$ elementos seguintes: $\{\}, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}$. O estado inicial é $\{1\}$ e os estados finais são: $\{2\}, \{1, 2\}, \{2, 3\}$ e $\{1, 2, 3\}$. Atendendo à definição dada para a função de transição deste novo autómato podemos traduzir o comportamento deste através da próxima tabela.

	$\{\}$	$\{1\}$	$\{2\}$	$\{3\}$	$\{1, 2\}$	$\{1, 3\}$	$\{2, 3\}$	$\{1, 2, 3\}$
a	$\{\}$	$\{1, 2\}$	$\{1\}$	$\{3\}$	$\{1, 2\}$	$\{1, 2, 3\}$	$\{1, 3\}$	$\{1, 2, 3\}$
b	$\{\}$	$\{2, 3\}$	$\{\}$	$\{2, 3\}$	$\{2, 3\}$	$\{2, 3\}$	$\{2, 3\}$	$\{2, 3\}$

A ilustração do comportamento do autómato determinístico construído, por meio de um diagrama de estados, encontra-se na Figura 2.8.

Como se observa este autómato é determinístico mas o seu diagrama de estados não possui um aspecto muito atractivo. Se repararmos, os estados $\{\}$, $\{2\}$ e $\{3\}$ não são atingidos, isto é, não são acessíveis. Dado um autómato determinístico $\mathcal{M} = \langle Q, \Sigma, \delta, q_0, F \rangle$, dizemos que um estado $q \in Q$ é *acessível* se existir $w \in \Sigma^*$ de forma que $\delta(q_0, w) = q$. No caso de todos os estados do autómato serem acessíveis, o autómato diz-se *acessível*.

Retirando os mencionados estados inacessíveis do autómato em causa, o correspondente diagrama de estados é apresentado como mostra a Figura 2.9.

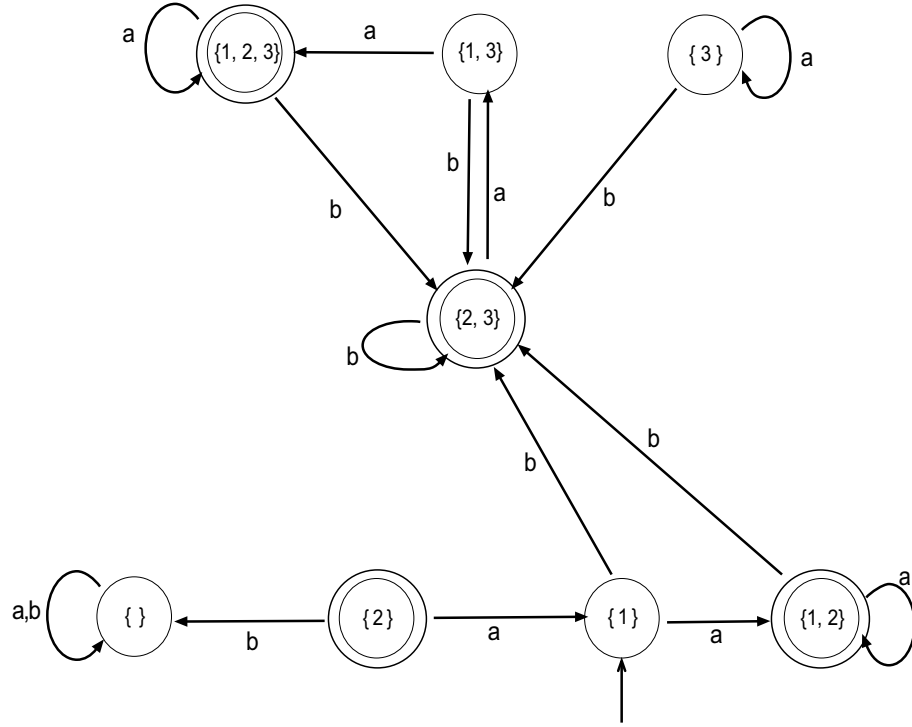


Figura 2.8:

Os exemplos apresentados encontram-se de forma mais detalhada em [20].

1.5 Operações com autómatos finitos

No que se segue consideramos os autómatos finitos e não-determinísticos $\mathcal{M} = \langle \Sigma, q_0, Q, F, \delta \rangle$ e $\mathcal{M}' = \langle \Sigma, q'_0, Q', F', \delta' \rangle$, tais que $Q \cap Q' = \emptyset$. Estes dois autómatos aceitam as linguagens $L(\mathcal{M})$ e $L(\mathcal{M}')$, respectivamente.

- Conexão em paralelo

A operação de conexão em paralelo entre autómatos está intimamente ligada à operação união entre as correspondentes linguagens dos autómatos. Daí que o autômato \mathcal{M}_u , resultante da conexão em paralelo dos autómatos \mathcal{M} e \mathcal{M}' , vai aceitar a linguagem $L(\mathcal{M}) \cup L(\mathcal{M}')$. Logo, o autômato \mathcal{M}_u aceita $w \in \Sigma^*$ se w for aceite por \mathcal{M} ou por \mathcal{M}' . Como assumimos que os autómatos são não-determinísticos, o autômato \mathcal{M}_u é também não determinístico e, ao receber a entrada w começa por escolher qual dos

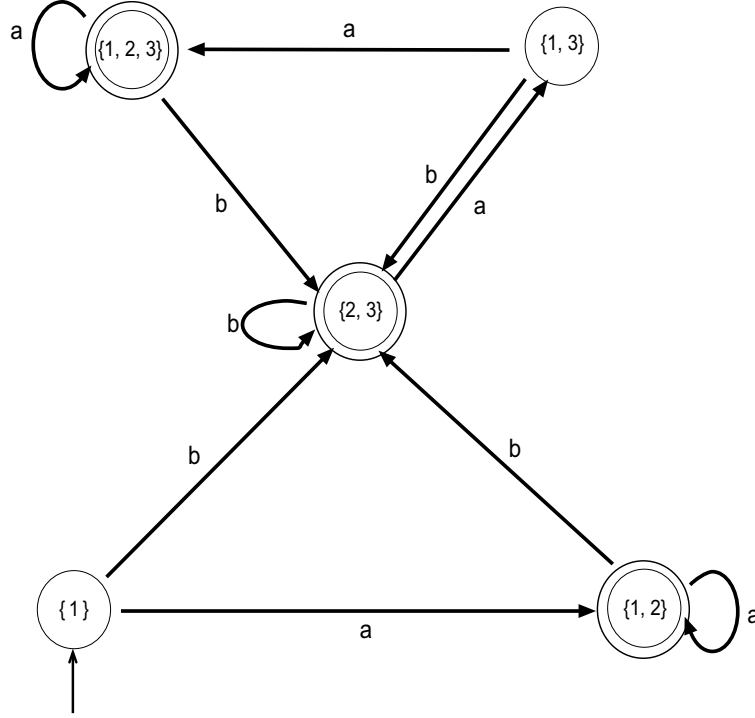


Figura 2.9:

autômatos, \mathcal{M} ou \mathcal{M}' , vai simular. De seguida executa o autômato escolhido, desde que w seja aceite pelo autômato eleito. Para tal é necessário criar um estado inicial, i , de \mathcal{M}_u , responsável pela transição para um dos estados iniciais, q_0 ou q'_0 . No entanto, é importante que após esta transição inicial, a entrada do autômato continue a ser w , daí que este novo estado i de \mathcal{M}_u assume, simultaneamente, as funções de q_0 , se \mathcal{M} foi escolhido ou de q'_0 , caso \mathcal{M}' tenha sido eleito. Para exemplificar, se \mathcal{M}_u escolheu simular \mathcal{M}' , então i realiza a transição com a entrada w como se fosse \mathcal{M}' com essa entrada. A figura 2.10 ilustra o que foi descrito.

A nova transição inicial está representada a tracejado para diferenciar das restantes transições. Como se observa na figura, os estados de \mathcal{M}_u são os mesmos de \mathcal{M} e \mathcal{M}' , acrescentando-se i . No que diz respeito aos estados finais, estes são a união dos estados finais de \mathcal{M} e \mathcal{M}' , ou seja, $F \cup F'$, desde que $q_0 \notin F$ ou $q'_0 \notin F'$. Se $q_0 \in F$ ou $q'_0 \in F'$, temos que impor que i seja estado final e, consequentemente, \mathcal{M}_u aceita ϵ . Formalmente,

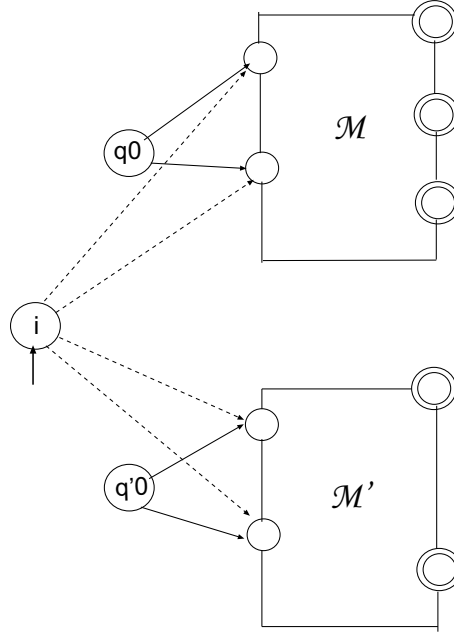


Figura 2.10:

$$\mathcal{M}_u = \langle \Sigma, i, Q_u, F_u, \delta_u \rangle$$

onde:

$$Q_u = i \cup Q \cup Q';$$

$$F_u = \begin{cases} \{i\} \cup F \cup F' & \text{se } q_0 \in F \text{ ou } q'_0 \in F' \\ F \cup F' & \text{se } q_0 \notin F \text{ e } q'_0 \notin F' \end{cases};$$

$$\delta_u(q, \sigma) = \begin{cases} \delta(q_0, \sigma) \cup \delta'(q'_0, \sigma) & \text{se } q = i \\ \delta(q, \sigma) & \text{se } q \in Q \\ \delta'(q, \sigma) & \text{se } q \in Q' \end{cases}$$

Exemplo 2.8. A conexão em paralelo dos autómatos \mathcal{M} e \mathcal{M}' cujos diagramas de estados se encontram nas Figuras 2.11 e 2.12, respectivamente, encontra-se representado na Figura 2.13.

O novo autômato foi construído tendo em conta a explicação teórica dada anteriormente.

- Conexão em série

Esta operação entre autómatos está relacionada com a operação concatenação entre as respectivas linguagens. Tencionamos construir um

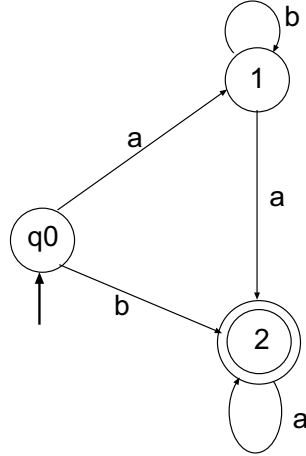


Figura 2.11:

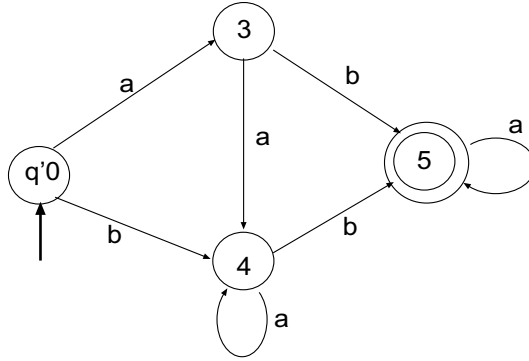


Figura 2.12:

autômato \mathcal{M}_c , resultante da conexão em série dos autômatos \mathcal{M} e \mathcal{M}' que aceite a linguagem $L(\mathcal{M}) \cdot L(\mathcal{M}')$. Dada uma palavra $w \in \Sigma^*$, o autômato \mathcal{M}_c aceita w se esta for composta por uma sequência inicial, chamada *prefixo*, aceite por \mathcal{M} e uma sequencial final, designada por *sufixo*, aceite por \mathcal{M}' . Esta ideia sugere que os autômatos \mathcal{M} e \mathcal{M}' sejam conectados um a seguir ao outro, isto é, em linha, apontando para um comportamento de \mathcal{M}_c igual a \mathcal{M} enquanto não atingir um estado final deste autômato. Posteriormente “imita” o comportamento de \mathcal{M}' até chegar a um seu estado final, caso o estado inicial de \mathcal{M}' não seja estado final de \mathcal{M}_c . Como não é possível estabelecer uma transição directa entre o estado final de \mathcal{M} e o estado inicial de \mathcal{M}' , uma vez que

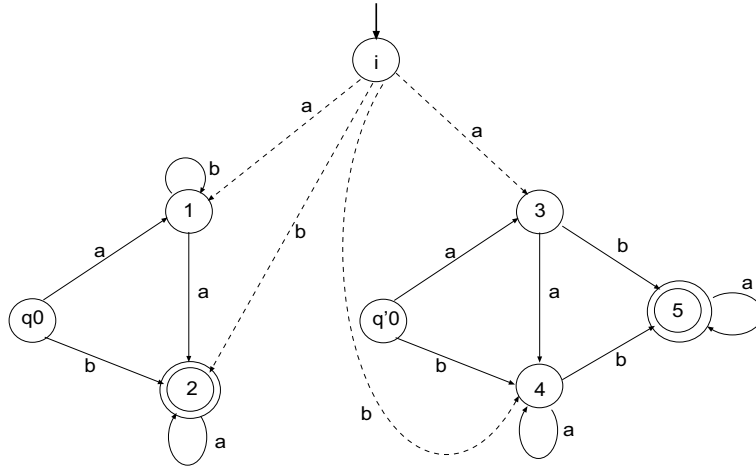


Figura 2.13:

os autômatos só transitam se houver entrada, torna-se necessário definir uma transição entre o estado final de \mathcal{M} e os sucessores do estado inicial de \mathcal{M}' . A situação encontra-se representada na figura 2.14.

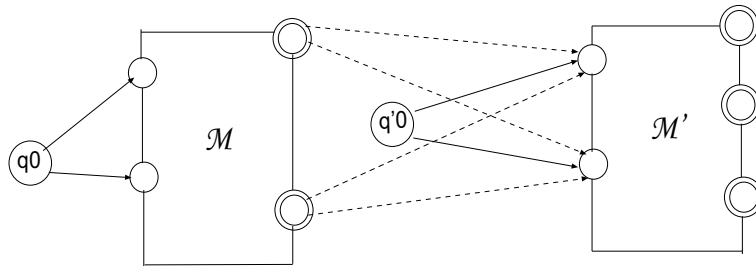


Figura 2.14:

Os estados finais de \mathcal{M} podem ser estados finais de \mathcal{M}_c . Mas tal só acontece se q'_0 for estado final de \mathcal{M}' . Caso contrário, os estados finais de \mathcal{M}_c são apenas os estados finais de \mathcal{M}' . Rigorosamente,

$$\mathcal{M}_c = \langle \Sigma, q_0, Q_c, F_c, \delta_c \rangle$$

onde:

$$Q_c = Q \cup Q';$$

$$F_c = \begin{cases} F \cup F' & \text{se } q'_0 \in F' \\ F' & \text{se } q'_0 \notin F' \end{cases};$$

$$\delta_c(q, \sigma) = \begin{cases} \delta(q, \sigma) & \text{se } q \in Q \setminus F \\ \delta(q, \sigma) \cup \delta'(q'_0, \sigma) & \text{se } q \in F \\ \delta'(q, \sigma) & \text{se } q \in Q' \end{cases}$$

Exemplo 2.9. A conexão em série dos autómatos \mathcal{M} e \mathcal{M}' cujos diagramas de estados se encontram nas Figuras 2.11 e 2.12, respectivamente, encontra-se representada na Figura 2.15. Novamente, o novo autômato é aplicação prática da teoria explicitada.

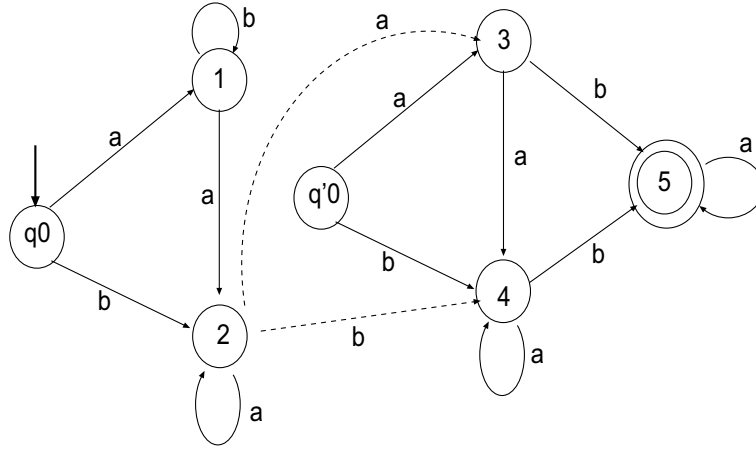


Figura 2.15:

- Fecho de Kleene

Acabamos de estudar o modo de obter um autômato por concatenação de outros dois. Uma outra operação entre autómatos finitos é a concatenação de um autômato com ele próprio. O objectivo é construir um autômato \mathcal{M}_K , fruto da concatenação de \mathcal{M} consigo mesmo, de tal modo que \mathcal{M}_K aceite a linguagem $L(\mathcal{M})^*$. A linguagem $L(\mathcal{M})^*$ é chamada o *fecho de Kleene* de $L(\mathcal{M})$, formalmente definida por $L(\mathcal{M})^* = \bigcup_{n=0}^{\infty} L(\mathcal{M})^n$, onde $L(\mathcal{M})^0 = \{\epsilon\}$ e $L(\mathcal{M})^n = L(\mathcal{M})^{n-1} \circ L(\mathcal{M})$.

A ideia mais imediata e intuitiva é construir o autômato \mathcal{M}_K conectando os seus estados finais com os seus estados iniciais. À semelhança do que aconteceu com a conexão em paralelo, há a necessidade de criarmos um novo estado inicial que faça a transição para os estados sucessores de q_0 e que seja final por forma a garantir a aceitação de ϵ . A ilustração do

comportamento de \mathcal{M}_K encontra-se na Figura 2.16. Mais uma vez, as novas transições encontram-se a tracejado.

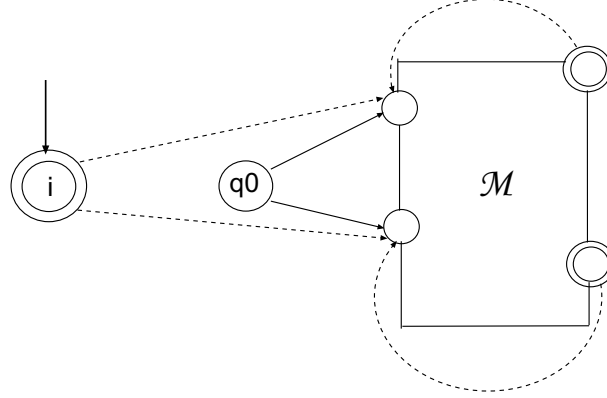


Figura 2.16:

De um modo mais formal,

$$\mathcal{M}_K = \langle \Sigma, i, Q_K, F_K, \delta_K \rangle$$

onde:

$$Q_K = \{i\} \cup Q;$$

$$F_K = \{i\} \cup F;$$

$$\delta_K(q, \sigma) = \begin{cases} \delta(q_0, \sigma) & \text{se } q = i \\ \delta(q, \sigma) & \text{se } q \in Q \setminus F. \\ \delta(q, \sigma) \cup \delta(q_0, \sigma) & \text{se } q \in F \end{cases}$$

Exemplo 2.10. Seguindo os passos para a construção do Fecho de Kleene de um autômato dado, apresentamos na Figura 2.17 o fecho de Kleene do autômato \mathcal{M} da Figura 2.11.

Os termos conexão em paralelo e conexão em série utilizados para nomear as operações entre autômatos são propositadamente empregues para estreitar a ligação que existe entre autômatos e circuitos eléctricos. No caso da conexão em paralelo, a entrada activa simultaneamente os dois circuitos. Na conexão em série, a entrada activa primeiro um dos circuitos e depois continua a activar o outro. No que diz respeito ao último caso, trata-se da conexão do circuito com ele próprio, onde a saída é total ou parcialmente utilizada como entrada. É vulgarmente conhecido como sistema de retro alimentação usado nos mecanismos de auto controle.

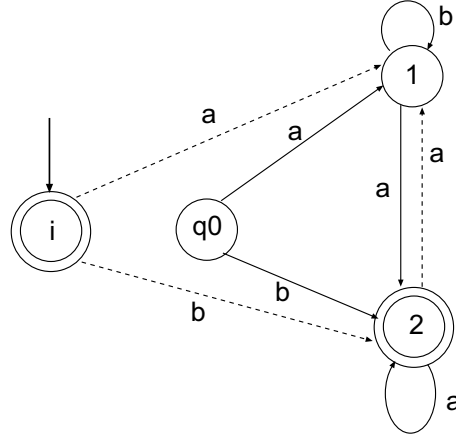


Figura 2.17:

Estas operações estão exhaustivamente descritas em [6] e [18].

2 Autômatos Probabilísticos

Na secção anterior falamos de autômatos em que o próximo estado é determinado única e exclusivamente pela função de transição, δ . Porém, existem autômatos nos quais o próximo estado é determinado de forma probabilística. Este tipo de autômatos são denominados *autômatos probabilísticos*. Por exemplo, os autômatos probabilísticos podem representar acontecimentos simples como é o caso do lançamento de uma moeda equilibrada, em que existe igual probabilidade de sair cara ou coroa, ou o lançamento de uma moeda não equilibrada em que a probabilidade de sair cada uma das faces é diferente. A ilustração destas duas situações encontra-se na Figura 2.18. Cada estado está representado por um nó no grafo, onde o *estado inicial* é representado por i . Cada seta tem a si associada a probabilidade de transição de um estado para outro.

A definição formal de autômato probabilístico é dada a seguir.

Definição 2.11. [Autômato Probabilístico] Um *autômato probabilístico* \mathcal{M}_p é um quintuplo $\langle \Sigma, Q, H, q_0, F \rangle$, tal que:

- $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_{|\Sigma|}\}$ é um conjunto finito de símbolos, denominado por alfabeto de entrada;

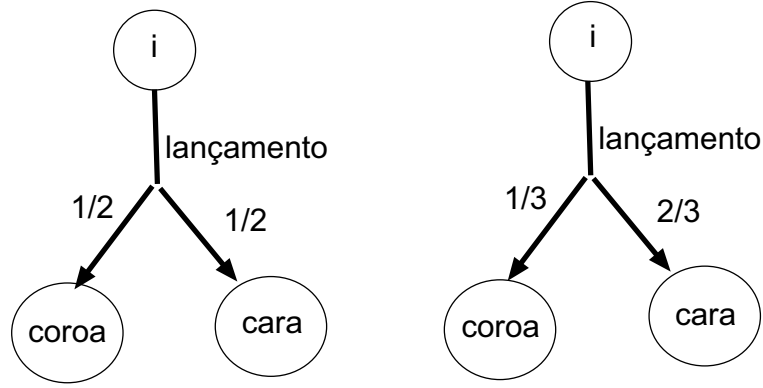


Figura 2.18: Representação da transição de estados no lançamento de uma moeda equilibrada e numa moeda não equilibrada.

- $Q = \{q_1, q_2, \dots, q_n\}$ é um conjunto finito de estados do autómato;
- H é uma aplicação de Σ no conjunto das matrizes estocásticas quadradas $n \times n$. Dado $a_m \in \Sigma$, $H(a_m) = [p_{ij}]$, onde $p_{ij} \geq 0$ é a probabilidade do autómato transitar do estado q_i para o estado q_j sob a entrada a_m , de tal forma que $\sum_{j=1}^n p_{ij} = 1$, para todo $i = 1, \dots, n$. Mais, a aplicação H pode ser estendida a Σ^* da seguinte maneira:
 - (i). $H(\epsilon) = I_n$, onde I_n é a matriz identidade $n \times n$;
 - (ii). $H(a_{m_1}, a_{m_2}, \dots, a_{m_k}) = H(a_{m_1})H(a_{m_2}) \cdots H(a_{m_k})$, com $k \geq 2$ e $a_{m_j} \in \Sigma$, para todo $j = 1, \dots, k$.
- $q_0 \in Q$ é o estado inicial do autómato;
- $F \subseteq Q$ é o conjunto dos estados finais do autómato.

Na Figura 2.19 encontra-se um exemplo de um autómato probabilístico. À semelhança do que acontecia com os autómatos, podemos também falar de *autómatos probabilísticos determinísticos* e *autómatos probabilísticos não-determinísticos*. A única diferença que existe entre estes dois tipos de autómatos probabilísticos reside no facto de nos autómatos probabilísticos do tipo determinístico, dado um estado actual e uma entrada, existe um único estado para o qual o autómato pode transitar. No caso dos autómatos probabilísticos do tipo

não-determinístico, dado um estado actual e uma entrada, existe um conjunto de estados para o qual o autómato pode transitar. Atendendo à definição de autómato probabilísticos dada anteriormente, é fácil verificar que no caso dos autómatos probabilísticos determinísticos, em cada linha da matriz H apenas existe um valor não nulo. Pelo contrário, cada linha da matriz H dos autómatos probabilísticos não determinísticos pode ter mais do que um valor diferente de zero. Observando o autómato da Figura 2.19 é fácil concluir que se trata de um exemplo de um autómato probabilístico determinístico, dado que, para cada entrada do alfabeto apenas existe uma saída a partir do estado actual do autómato.

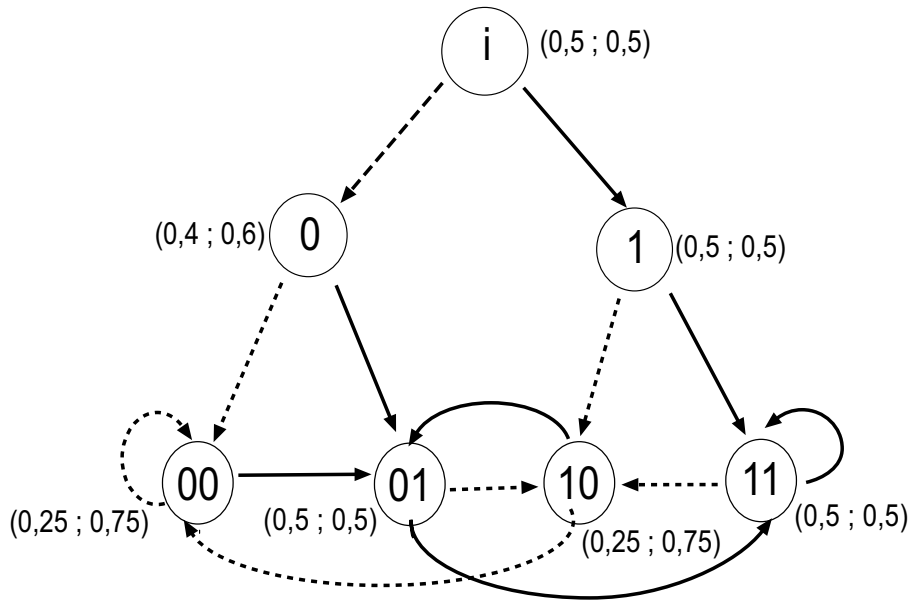


Figura 2.19: Exemplo de um autómato probabilístico finito.

Cada estado do autómato é representado por um nó do grafo e as setas que saem de cada estado são itequetadas ou diferenciadas de acordo com os símbolos do alfabeto do autómato. Os estados têm a eles associado um vector de probabilidades, cujas componentes correspondem à probabilidade do próximo símbolo, para cada um dos símbolos do alfabeto.

No caso da Figura 2.19, o alfabeto, Σ , é unicamente formado pelos símbolos 0 e 1 . Os estados do autómato, Q , é o conjunto $\{i, 0, 1, 00, 01, 10, 11\}$. Por exemplo, o estado 10 é resultado da seguinte sequência de símbolos: 1 como o

primeiro valor e 0 como o último valor. Como se pode verificar, cada estado tem a si associado um vector de probabilidades. Tomando o estado 10, observamos que este tem a ele associado o vector de probabilidades $(0, 25; 0, 75)$, o que significa que 0 pode ser o próximo símbolo a aparecer com probabilidade de 0,25 e o símbolo 1 tem probabilidade 0.75 de ser o próximo a surgir. As setas contínuas representam as transições correspondentes ao símbolo 1 e as setas descontínuas as transições do símbolo 0. Se, por exemplo, o autómato se encontra no estado 10 e o próximo símbolo é o 0, então o próximo estado é o 00; mas se o próximo símbolo é o 1, o próximo estado é o 01.

Mais informação acerca dos autómatos probabilísticos pode ser encontrada em [5], [26] e [29].

2.1 Linguagem aceite por um autómato probabilístico

A linguagem aceite por um autómato probabilístico, \mathcal{M}_p , é a linguagem $\mathcal{L}(\mathcal{M}_p) = \{(w; p(w)) : w \in \Sigma^*, p(w) = \pi_0 H(w) \pi_F > 0\}$, onde π_0 é um vector linha n -dimensional, cuja i -ésima componente é igual a um se $q_i = q_0$ e zero caso contrário; π_F é um vector coluna n -dimensional cuja j -ésima componente é igual a um se $q_j \in F$ e zero caso contrário.

Por vezes define-se no autómato probabilístico um limite inferior λ (probabilidade mínima exigida para que se considere uma palavra aceite), $0 \leq \lambda \leq 1$. A linguagem aceite para este autómato probabilístico é $\mathcal{L}(\mathcal{M}_p, \lambda) = \{(w; p(w)) : w \in \Sigma^*, p(w) = \pi_0 H(w) \pi_F > \lambda\}$, onde π_0 e π_F são os vectores definidos anteriormente [27].

Os autómatos probabilísticos reconhecem, exactamente, a classe das linguagens regulares pesadas. Esta classe de linguagens contém estritamente a classe das linguagens regulares, [19] e [27]. Em [19] é dado o exemplo de uma gramática regular pesada que gera uma linguagem livre de contexto que não é regular.

2.2 Autómatos Probabilísticos e Redes Neurais

Começamos por esclarecer o que se entende por redes neurais. O cérebro humano é formado por, aproximadamente, 10 biliões de neurónios, conectados por sinapses e, juntos, constituem uma grande rede, chamada *Rede Neuronal*.

A primeira tentativa de construção de uma máquina inspirada no cérebro humano surgiu em artigos de McCulloch e Pitts, por volta de 1943. Apareceu assim a denominada *Neuro Computação*. Muitos outros trabalhos se seguiram nesta área ao longo de muitos anos. Realçamos aqui o trabalho de Donald Hebb em 1949, com a publicação do livro intitulado “The Organization of Behavior”, no qual Hebb propõe uma lei de aprendizagem para os neurónios. Em 1951 surge o primeiro neuro computador, chamado Snark, pela mão de Marvin Minsky. No entanto, o primeiro neuro computador a ter sucesso foi Mark I Perceptron que apareceu em 1957 e 1958, criado por Frank Rosenblatt, Charles Wightman e outros. Atendendo à profundidade dos seus estudos, Rosenblatt é visto como o fundador da neuro computação tal como a conhecemos nos dias de hoje. Nos anos seguintes assistiu-se a um período de numerosas investigações nesta área, período este que culminou com uma fase de pesquisa silenciosa que teve lugar entre 1967 e 1982. Os anos 1980 voltaram a ser palco de várias publicações, onde se destaca o físico John Hopfield. Em 1987 ocorreu, em São Francisco, a primeira Conferência de Redes Neurais e foi criada a International Neural Networks Society (INNS). Fundaram-se também o INNS journal em 1989, seguindo-se o Neural Computation journal.

Com a evolução crescente da tecnologia, tem sido desejo do Homem a criação de máquinas que possam operar independentemente do controle humano. Pretende-se a criação de sistemas computacionais mais eficientes e inteligentes do que os actuais.

As redes neuronais artificiais (RNA) constituem um método para solucionar problemas de inteligência artificial, permitindo a construção de sistemas que simulem o cérebro humano. São técnicas computacionais que apresentam um modelo inspirado na estrutura neuronal dos organismos inteligentes.

As redes neuronais podem ser entendidas como um grafo, cujos nós são chamados *elementos de processamento*; as arestas são *conexões*, que funcionam como caminhos de condução de sinais. Estas estruturas podem ou não possuir memória local e podem conter um qualquer número de conexões de saída.

De acordo com as características específicas de cada rede neuronal, estas podem ser distribuídas por diferentes classes.

No que se segue faremos a ligação entre a classe das redes neuronais com pesos, GSSW (General Single - layer Sequential Weighted) e os autómatos pro-

probabilísticos. Este tipo de redes neuronais são implementadas com neurónios do tipo pRAM (probabilistic random access memory) ou com neurónios do tipo MPLN (Multi - valued Probabilistic Logic Nodes). Este último tipo de neurónios são uma extensão dos PLNs (Probabilistic Logis Nodes). O que pretendemos é referir um algoritmo que estabelece a equivalência, em termos computacionais, entre redes neuronais GSSW e os autómatos probabilísticos. A definição de autómato probabilístico usada será a que foi dada anteriormente. Porém, para simplificação de provas, e sem perda de generalidade, os autómatos probabilísticos que iremos considerar terão, no máximo, duas setas de saída etiquetadas com o mesmo símbolo, em cada estado. Note-se que é trivialmente possível transformar um qualquer autómato probabilístico, num outro equivalente com esta propriedade. Para tal, basta introduzir estados auxiliares e introduzir transições vazias. As redes neuronais GSSW têm uma definição semelhante aos autómatos probabilísticos e como é nosso propósito estabelecer a equivalência entre estas redes e os autómatos probabilísticos, vamos assumir que as redes possuem também a propriedade atrás mencionada para os autómatos, isto é, terão no máximo dois estados actuais em cada passo. Para tal, o conjunto de valor possíveis para os estados é reduzido a seis valores diferentes $\{p_1, p_2, \dots, p_6\}$, onde $p_i \in [0, 1], i = 1, \dots, 6$. Estes valores estão relacionados com o conjunto de vectores binários $\{(0); (1); (0, 0); (0, 1); (1, 0); (1, 1)\}$. A rede neuronal GSSW pode ser formalmente definida como se segue:

Definição 2.12. [Rede Neuronal GSSW]

Uma rede neuronal GSSW é um 7-uplo $N = \langle X, U, y, f, p, x_0, R \rangle$, onde:

- (i). $X = \{0, 1\}^{n_x}$ é o conjunto de estados da rede neuronal;
- (ii). $U = \{0, 1\}^{n_u}$ define o conjunto dos possíveis vectores de entrada;
- (iii). $y = [0, 1]$ é a saída da rede neuronal;
- (iv). f é a função de transição;
- (v). p é a função de transição de probabilidade;
- (vi). x_0 é o estado inicial da rede neuronal;

(vii). $R \subseteq X$ é o conjunto dos estados aceites pela rede neuronal.

O teorema seguinte estabelece a ligação entre os autómatos probabilísticos e as redes neuronais GSSW.

Teorema 2.13. *Seja $\mathcal{M}_p = \langle \Sigma, Q, H, q_0, F \rangle$ um autómato probabilístico com limite λ . Então existe uma rede neuronal GSSW que implementa \mathcal{M}_p .*

A demonstração deste teorema fornece um procedimento para a transformação de um autómato probabilístico numa rede neuronal GSSW. A ideia essencial da prova do teorema é estabelecer uma correspondência entre cada componente do autómato probabilístico com cada um dos constituintes da rede neuronal. A prova detalhada e completa do teorema pode ser consultada em [27].

2.3 Autómatos Probabilísticos e Modelos Escondidos de Markov

Os HMM (Hidden Markov Models) foram descritos, pela primeira vez, entre o final da década de 1960 e o início da década de 1970, sendo utilizados em reconhecimento de palavras a partir de 1970.

Verifica-se que, nos últimos anos, os HMM têm sido amplamente aplicados em várias áreas, nomeadamente o reconhecimento de voz, a modelagem de linguagem, o reconhecimento de palavras manuscritas, a verificação on-line de assinaturas, a detecção de falhas em sistemas dinâmicos, entre outras. Diz-se que o HMM é um processo duplamente estocástico, uma vez que possui um processo estocástico não visível (daí a designação de escondido), mas que pode ser observável através de um outro processo estocástico que produz uma sequência de observações visíveis. O processo estocástico escondido é formado por um conjunto de estados, conectados por transições, às quais estão associadas probabilidades. O processo estocástico visível é constituído por um conjunto de saídas ou observações, emitidas pelos estados invisíveis, de acordo com os valores da função de densidade de probabilidade(fdp), [9].

A Figura 2.20 fornece uma visão global do funcionamento de um HMM, onde cada forma oval representa uma variável aleatória. A linha de cima é

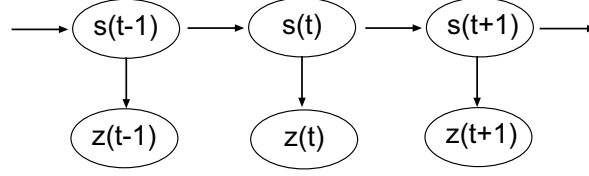


Figura 2.20: Esquema de funcionamento de um modelo escondido de Markov.

composta pelas variáveis aleatórias escondidas que tomam o valor $s(t)$, no instante t . A linha de baixo é formada pelas saídas observáveis que tomam o valor $z(t)$ no instante t . O valor da variável escondida $s(t)$ depende unicamente do valor da variável escondida anterior, $s(t-1)$. A esta propriedade chamamos *propriedade de Markov*. Analogamente, o valor da saída $z(t)$ depende exclusivamente do valor tomado pelo estado $s(t)$. De acordo com as características da fdp, os HMM's podem ser classificados em discretos, contínuos e semi-contínuos. A grande vantagem da utilização de HMM reside na sua natureza probabilística, adequada para sinais corrompidos por ruídos, tal como a fala ou a escrita, assim como na sua fundamentação teórica, ao nível da existência de algoritmos eficientes no ajustamento dos parâmetros ao modelo, através de métodos iterativos, [9].

Formalmente, um HMM pode ser definido como se segue, [30].

Definição 2.14. [Hidden Markov Model(HMM)] Um *hidden Markov model* é um quádruplo $\mathcal{P} = \langle \Sigma, Q, A, B, \tau \rangle$, onde:

- (i). Σ é um alfabeto finito;
- (ii). Q é um conjunto finito de estados;
- (iii). $A : Q \times Q \rightarrow [0, 1]$ é a função que define a probabilidade de cada transição, de tal modo que

$$\forall q \in Q, \sum_{q' \in Q} A(q, q') = 1;$$

- (iv). $B : Q \times \Sigma \rightarrow [0, 1]$ é a função que determina a probabilidade saída de

cada símbolo do alfabeto em cada estado, tal que

$$\forall q \in Q, \sum_{a \in \Sigma} B(q, a) = 1;$$

(v). $\tau : Q \rightarrow [0, 1]$ é a função que define a probabilidade inicial para cada estado, tal que

$$\sum_{q \in Q} \tau(q) = 1.$$

Pierre Dupont [30], apresenta uma demonstração construtiva do seguinte:

Teorema 2.15. *HMMs são equivalentes a autómatos probabilísticos sem probabilidades finais.*

A prova apresentada é uma prova construtiva, onde Dupont mostra as seguintes implicações $PNFA \Rightarrow HMMT \Rightarrow HMM \Rightarrow PNFA$, onde:

- (i). $PNFA$ é um autômato probabilístico não-determinístico sem probabilidades finais, isto é, um autômato probabilístico não-determinístico em que o conjunto dos estados finais do autômato é vazio;
- (ii). $HMMT$ (hidden Markov model with transition emission) é um HMM em que a aplicação B está definida da forma que se segue:

$$B : Q \times \Sigma \times Q \rightarrow [0, 1], \forall q, q' \in Q, \sum_{a \in \Sigma} B(q, a, q') = \begin{cases} 1 & \text{se } A(q, q') > 0 \\ 0 & \text{noutros casos} \end{cases}.$$

No entanto, já em 1992, Abe e Warmuth, [2], demonstraram que qualquer HMM pode ser transformado num autômato probabilístico com o mesmo número de estados. Para tal, os autores afirmam que o conjunto de estados do HMM passa a ser o conjunto de estados do autômato; o alfabeto do HMM passa a ser o alfabeto do autômato e constroem as aplicações do autômato a partir das aplicações do HMM.

3 Aplicações

Tal como em qualquer outra área da matemática, a grande motivação para o crescente estudo e investigação em teoria dos autómatos são as aplicações práticas.

Os autómatos finitos são particularmente utilizados na busca de padrões, isto é, na procura de uma palavra ou conjunto de palavras num texto. Um texto é uma sequência de símbolos como por exemplo uma página de um livro ou uma cadeia de ADN (ácido desoxirribonucleico) ou em inglês DNA (deoxyribonucleic acid). Este tipo de problema, aparentemente simples, tem sido alvo de grandes investigações desde 1970, com o desenvolvimento de algoritmos para a sua resolução. Muitos dos algoritmos encontrados são matematicamente suportados por resultados da teoria dos autómatos e encontram-se, actualmente, aplicados em compiladores, editores de texto e sistemas de manipulação de arquivos. A descrição mais pormenorizada de alguns algoritmos de busca de padrões pode ser encontrada em [10].

A partir de 1970 apareceram, pela mão do matemático Jonh Conway e através dos seu jogo *game of life*, os *autómatos celulares*. Um *autómato celular* é uma estrutura formada por células, as quais são autómatos. Podem-se construir autómatos celulares de várias dimensões. Um autómato celular de uma dimensão é uma sequência infinita de autómatos justapostos em linha. O autómato celular de duas dimensões é formado por autómatos colocados lado a lado de forma a preencherem o plano, [32]. Este tipo de autómatos tem sido utilizado com êxito no estudo e simulação de sistemas dinâmicos e complexos na biologia e na física. Um exemplo concreto é o caso da modelação da propagação dos fogos florestais e a movimentação das nuvens, permitindo a previsão do comportamento das nuvens num curto período de tempo ou traçar a forma de propagação de uma epidemia provocada por um vírus ao longo do tempo. Estes exemplos encontram-se de forma detalhada em [31].

Com a crescente complexidade dos circuitos e sistemas electrónicos, houve a necessidade de se desenvolver técnicas mais eficazes e menos dispendiosas para fazer face a este desenvolvimento. Ao longo dos anos muitos problemas relacionados com a síntese, análise e optimização de circuitos sequenciais têm sido aproximados por conceitos da teoria dos autómatos. Mais ainda, têm-se verificado a importância e utilidade em introduzir um comportamento probabilístico aos autómatos, ou seja, usar como modelo matemático os autómatos probabilísticos, [11]. Este género de autómatos tem encontrado grande aplicabilidade a nível da biologia computacional, nomeadamente, no alinhamento de sequência biológicas, na procura de padrões nessas mesmas sequências e

na análise da estrutura secundária do ARN (ácido ribonucleico) ou em inglês RNA (ribonucleic acid), [24]. Os autómatos probabilísticos são também utilizados como meio de análise de séries temporais. No artigo [26] encontramos um exemplo concreto do uso de um autômato probabilístico na modelação de parâmetros e previsão de novos valores para esses parâmetros relacionados com o clima. Os autómatos probabilísticos constituem a nossa principal fonte de inspiração para o desenho da proposta de sistema inteligente de ensino assistido.

Capítulo 3

Ensino Assistido por computador

“Com o advento do computador, tornou-se claro que os módulos do material de instrução poderiam passar a ser apresentados com grande flexibilidade. Assim, durante o início dos anos sessenta, foram criados diversos programas informáticos de instrução programada e começou a popularizar-se a expressão “ensino assistido por computador” (EAC) ou “computer-aided instruction” (CAI). Aliás, os primeiros anos do processo de integração dos computadores nas escolas ficaram muito marcados pela tentativa da sua utilização de modo a melhorar a eficácia do acto de ensinar.” [21]

Um computador é actualmente, e em grossos modos, uma máquina capaz de processar grandes quantidades de informação num curto espaço de tempo. No entanto, a palavra computador é frequentemente usada para denominar determinado tipo de dispositivo electrónico digital. O computador digital é uma máquina que efectua operações de aritmética, de transferência e de decisão sobre uma certa quantidade de informação apresentada de forma conveniente e devolve o resultado dessas operações numa forma considerada adequada. Outro tipo de computadores são os do tipo analógico que servem essencialmente para simular o comportamento de sistemas físicos, como por exemplo, o braço de um robot, moldados por modelos matemáticos. São fundamentalmente utilizados quando o ensaio no sistema real se torna difícil ou impossível, devido a questões

relacionadas com o perigo, o ambiente ou o custo.

Na secção que se segue abordaremos, numa perspectiva histórica, a evolução dos computadores digitais, simplesmente conhecidos por computadores.

1 O computador ao longo dos tempos

Podemos dizer que o ábaco, criado por volta do ano 3000 a. C., deu início ao processo de evolução de dispositivos de tratamento de informação. Depois do ábaco, muitas foram as máquinas criadas para a realização de cálculos repetitivos. A primeira máquina de somar que se conhece foi criada em 1642 por Blaise Pascal (1623 - 1662), filósofo e cientista francês. Era um computador mecânico capaz de realizar adições e subtracções através do movimento de rodas dentadas para a entrada dos operandos. Outras máquinas computadoras surgiram pela mão de Gottfried Leibniz (1646-1716), Charles Babbage (1791-1871), entre outros. Mas é a partir da última década do século XIX que se iniciam mudanças progressivas e profundas nas técnicas de construção dos computadores. A título de curiosidade, referimos que os dados relativos aos censos de 1880 levaram cerca de sete anos a serem tratados manualmente. Com o aumento demográfico existente, estimava-se que os resultados dos censos de 1890 levassem mais de dez anos a serem analisados. Foi aí que surgiu uma máquina criada por Herman Hollerith que reduziu o tempo para três anos. A máquina criada por Hollerith era capaz de ordenar os cartões do censo por diferentes categorias, utilizando sensores eléctricos para ler os milhões de cartões. Com o intento de divulgar e, porventura, vender esta máquina, Hollerith fundou a International Business Machines Corporation (IBM). Com o apoio da IBM, Dr. Howard Aiken, professor da universidade de Harvard, projectou uma máquina de cálculo automático que entrou em funcionamento em 1944, com o nome Harvard Mark I Automatic Sequence Controlled Calculator. Era uma enorme calculadora mecânica e eléctrica de quinze toneladas que esteve durante quinze anos a produzir tabelas para fins científicos, de engenharia e militares.

Em 1946 despontou o primeiro computador electrónico digital chamado ENIAC (Electronic Numerical Integrator and Calculator). Foi construído entre 1943 e 1946 na universidade de Pensylvania sob orientação de John W. Mauchly e J. Presper Eckert, procurando satisfazer a necessidade do exército

dos Estados Unidos da América no que concerne a tabelas de balística. Possuía válvulas electrónicas em vez de componentes mecânicas, o que lhe proporcionava uma maior velocidade na realização das tarefas. No entanto, não tinha qualquer tipo de memória interna.

O aperfeiçoamento contínuo dos processos de construção dos circuitos que constituem a base de funcionamento de qualquer computador tornou possível a passagem das válvulas para transístores e destes a circuitos integrados com um crescente grau de complexidade até chegar ao que vulgarmente chamamos de *chip*. Entre 1955 e 1964 começaram a aparecer linguagens relativamente simples para o Homem e que podiam ser facilmente traduzidas para a linguagem da máquina, as denominadas linguagens de alto nível, como por exemplo, FORTRAN, ALGOL e COBOL. Os computadores desta época já eram capazes de executar mais do que um programa em simultâneo, o que obrigou à introdução de um gestor das tarefas da máquina, chamado *sistema operativo*. Após 1965 houve tendência para diminuir o tamanho dos equipamentos. Entre 1964 e 1975 apareceram os computadores de alto desempenho, caracterizados pelo tratamento rápido dos dados. Contudo, este tipo de computadores era bastante caro. Com o passar do tempo a tecnologia tornou-se progressivamente mais barata surgindo uma nova gama de computadores, os computadores pessoais (PC - personal computers). São computadores relativamente pequenos, de baixo custo, feitos para estar numa secretária ou, quando apresentados de forma mais compacta, podem ser transportados (os PC portáteis).

A revista *Time*, proclamou 1982 como o ano do computador devido ao forte contributo dos computadores pessoais no trabalho humano. Este foi também o ano em que a Microsoft desenvolveu o sistema operativo MS-DOS para IBMs compatíveis. Dois anos depois a companhia Apple lançou o computador Macintosh, caracterizado pela sua interface gráfica. Em 1990 a Microsoft surge com o Windows 3.0 para fazer frente à Macintosh.

Nos últimos anos assistiu-se a uma nova evolução na forma de comunicar entre o computador e o utilizador. Um “menu” de possibilidades é apresentado ao utilizador em “janelas”, muitas vezes com ícones e símbolos sugestivos. A interacção é estabelecida através de um pequeno objecto que deslocamos sobre a mesa e que controla a posição do ponteiro no ecrã, ao qual chamamos “rato”. Tem-se, assim, uma interacção essencialmente visual e particularmente

envolvente.

O próximo passo foi a realidade virtual. Com esta tecnologia os computadores de hoje são capazes de gerar ambientes a três dimensões, interactivos, dando a sensação de que estamos noutro mundo, noutro local.

Em relativamente pouco tempo passamos das máquinas dispendiosíssimas e de grandes dimensões que apenas estavam ao alcance das grandes empresas e que funcionavam somente como uma espécie de super-calculadoras, para aparelhos pequenos e portáteis já acessíveis à maior parte dos camadas sociais, com uma grande capacidade de processamento e capazes de realizar tarefas diversificadas. Tudo leva a crer que o computador do futuro entrará em interacção com o utilizador através da linguagem comum e da linguagem gráfica.

2 O computador na sociedade

Em tempos que já lá vão, a agricultura marcou o início da civilização. Desde então, todas as alterações na dinâmica da sociedade são baseadas em inovações, como por exemplo, o desenvolvimento do comércio aquando das navegações marítimas, o desenvolvimento industrial baseado no carvão e no aço e, mais recentemente, na electricidade. Depois das revoluções agrícola e industrial, chega a vez da revolução da informação. De acordo com Toffler [34], estamos presentemente a assistir à terceira revolução tecnológica da humanidade: a passagem de uma sociedade industrial para uma sociedade de informação, à qual chama *Terceira Vaga*. A sociedade actual vive momentos de mudança. As novas tecnologias, e concretamente o computador, são parte integrante da sociedade em que vivemos. Nos dias de hoje, o computador ocupa um lugar primordial na sociedade. Analisemos alguns exemplos concretos dessa presença tão marcante. Toda a informação bancária é realizada através de computadores. As próprias pessoas, ao utilizarem uma caixa multibanco ou um qualquer computador, estão a comunicar com o banco por meio de um computador. Toda a informação referente aos habitantes de Portugal na segurança social, finanças e outros organismos do estado é feita e arquivada por computadores que actualmente permitem o tão popular cruzamento de dados. Na maioria dos supermercados, nos correios e em outros locais de comércio, as registadoras manuais foram substituídas por leitores ópticos sensíveis à banda de código de

barras existentes nas embalagens dos produtos. Esses leitores ópticos, ligados a um computador dão a indicação do preço do produto, procedendo simultaneamente à gestão do stock. Em praticamente todas as empresas a informatização é um bem adquirido, não só a nível administrativo, mas também ao nível do equipamento. Em suma, os computadores estão presentes nas mais diversas actividades sociais, empresariais e industriais. O computador está ainda presente na vida diária de cada um e já invadiu os lares portugueses. Segundo um estudo da consultora Pyramide, realizado a dez países europeus, 42% das casas portuguesas possuem computador. Apesar desta taxa atingir quase os 50%, ainda está muito abaixo da média europeia, 70,1%. De acordo com o mesmo estudo, os três países com maior taxa são o Reino Unido com 88%, Holanda com 87% e Áustria com 86%. A Suíça ocupa o quarto lugar com 75%, seguindo-se a Alemanha com 74% e a Bélgica com 73%. Abaixo da média europeia encontram-se a Itália com 61%, a França com 59% e a Espanha com 56%.

Está ultrapassada a ideia de que os computadores são inacessíveis à maior parte das pessoas, quer a nível financeiro, quer a nível do seu manuseamento. No que diz respeito ao nível financeiro, é bem visível a diminuição do custo dos computadores ao longo dos tempos, o que faz com que a maior parte das pessoas possam adquirir um computador para o seu lar. Quanto ao nível do seu manuseamento, temos um grande número de cursos de informática destinados a crianças, adultos e até mesmo a idosos. Não pára de crescer o número de iniciativas que visam dar a conhecer o computador às populações mais idosas. A 9 de Dezembro de 2005 era notícia que estava a decorrer, desde o dia 16 de Janeiro desse ano, o curso de informática para idosos. A iniciativa era da Junta de Freguesia de S. João da Talha e contava com o apoio do Centro de Divulgação da Tecnologia e Informação (CDTI). Mais perto, no concelho de Estarreja, também existem cursos de informática para idosos com o apoio do CDTI e os lares de terceira idade estão apetrechados de computadores ligados à Internet. Estes são apenas dois exemplos que resolvemos apresentar dos muitos que existem por todo o país e que podem ser consultados em www.jf-sjoaodatalha.pt e www.cm-estorreja.pt, respectivamente.

É urgente promover o acesso à info-alfabetização e à info-competência, como forma de combater não só possíveis discriminações sociais como também

garantir a sobrevivência das organizações. É certo que quem não souber lidar com as novas tecnologias da comunicação, concretamente o computador, é um analfabeto funcional. Nos dias que correm são vários os sectores que requerem dos seus empregados conhecimentos básicos de informática. Este novo conceito de sociedade, a *sociedade da informação*, teve a sua origem pouco antes do início do século XXI. Desde a disponibilização de recursos multimédia, passando pela crescente capacidade de armazenar e gerir grandes quantidades de informação, conseguimos já ligar o computador pessoal a outros computadores unicamente através de uma linha telefónica. Esta capacidade derrubou limites geográficos e culturais, deixando a distância de ser um factor decisivo e limitador. Neste propósito assenta uma das maiores conquistas desta nova sociedade, a Internet. Ainda de acordo com o estudo da consultora Pyramide, 60% dos computadores dos lares portugueses estão ligados à Internet por banda larga, colocando Portugal no primeiro lugar da lista. Seguem-se a Itália com 58% e a Suíça com 57%. A Alemanha está em último lugar com 25%, a Áustria e o Reino Unido em penúltimo lugar com 33%.

A preparação dos cidadãos para a nova era da informação é o grande desafio que se coloca. Assim sendo, verificou-se na última década do século XX um crescente número de iniciativas europeias e nacionais que procuraram dar resposta a este desafio. Em 1993, a Comissão Europeia lança o Livro Branco “Crescimento, competitividade e emprego - os desafios e as pistas para entrar no século XXI”, onde reconhece a importância da instauração da sociedade de informação na Europa. Em 1996, apresenta o Livro Verde “Viver e trabalhar na sociedade da informação: prioridade à dimensão humana” destinado a aprofundar o diálogo político, social e civil acerca dos aspectos sociais mais importantes da sociedade de informação. Também em 1996, a Comissão Europeia publica a comunicação “Aprender na sociedade de informação. Plano de acção para uma iniciativa europeia na educação (1996-1998)”, onde são definidas quatro linhas orientadoras para o apoio de iniciativas locais e nacionais. Em 1997, o Conselho da União Europeia aceita a decisão que adopta um Programa Comunitário Plurianual de Incentivo à Instauração da Sociedade de Informação na Europa (Sociedade de Informação), onde são estabelecidos objectivos e medidas que procuram a implementação da Sociedade de Informação na Europa.

3 O computador na escola

A exigência da sociedade actual em constante desenvolvimento e com a invasão das novas tecnologias, provocou a necessidade de se repensar o processo de ensino-aprendizagem, numa perspectiva de alteração dos métodos tradicionais de ensino. Em vez de obrigar o aluno a reter grandes quantidades de informação, muitas vezes sem ligação entre os conhecimentos ministrados, é urgente levar o aluno a compreender novas situações de modo a poder reagir perante o imprevisto. Segundo Durkheim, “a educação é a acção exercida pelas gerações adultas sobre aquelas que ainda não estão maduras para a vida social. Tem por objecto suscitar e desenvolver na criança um certo número de estados físicos, intelectuais e morais que lhe exigem a sociedade política no seu conjunto e o meio especial ao qual está particularmente destinada.”[28]. É portanto importante ajudar o aluno a adquirir competências básicas que o permitam adaptar-se à mudança. O ensino de hoje tem que ensinar o aluno a pensar e motivá-lo para uma aprendizagem ao longo da vida, passando o aluno a ter um papel activo no processo de ensino-aprendizagem. Assim, ao contrário do que acontecia anteriormente, o aluno deixa de ser um mero espectador e torna-se participante activo no seu processo de aprendizagem. Colocando o aluno no centro de todo o processo, o papel do professor é naturalmente alterado. Desta forma, o professor deixa de ser o único detentor do saber e passa a ser o gestor das diversas actividades.

No início, quando os computadores ganharam portabilidade e o seu custo começou a descer, acreditou-se no sucesso do ensino assistido por computador e houve mesmo quem pensasse na possível substituição dos professores por máquinas de ensinar. Citando Ponte, “Mais do que a visão do computador como substituto do professor, importa reter a ideia do computador como instrumento de consulta e de trabalho para assistir os alunos nas suas investigações e nos seus projectos.”[8]. É claro que o computador não tem noção com quem está a lidar e nem sequer tem sensibilidade humana. Daí que esta nova escola não pode menosprezar ou diminuir a importância do papel do professor. Ainda Ponte, citando Jerry Jinks explica como este via uma “nova” aula de História. Jerry Jinks sugeria que, através de um sistema de projecção tridimensional, se tivesse, por exemplo, a visita de D. Afonso Henriques o qual

poderia responder às perguntas dos alunos como se fosse um actor de carne e osso. Propunha também aplicações do género a aulas de outras disciplinas, como por exemplo a disciplina de Geografia onde os alunos podiam visitar os países e não só ler livros ou artigos dos países em que estavam interessados, ou simplesmente ouvir passivamente a descrição feita pelo professor. Devemos encarar o computador como um auxílio tecnológico e um parceiro do professor neste árduo processo de ensino-aprendizagem. Como afirma Ponte, “Será a combinação dos dois, ambos no máximo das suas possibilidades, que constituirá a equipa pedagógica do futuro.” [8].

4 Introdução das novas tecnologias nas escolas de Portugal

O desenvolvimento acelerado das novas tecnologias da informação e comunicação e os seus reflexos a nível económico, social, cultural e educacional exigiu a tomada de medidas por parte das entidades e instituições governamentais no sentido de enfrentarem os desafios que se colocam com estas mudanças. Uma boa fonte de informação é o Relatório para a Unesco da comissão internacional sobre educação para o século XXI - Educação, um tesouro a descobrir, de 1996. A nível nacional, em Abril de 1997, é aprovado o “Livro Verde para a sociedade de informação em Portugal”, documento no qual são apresentadas medidas a promover pelo Estado por forma a permitir o acesso e plena integração dos cidadãos na sociedade de informação. Mais próximo das escolas foram implementados alguns projectos sobre os quais faremos uma concisa menção de seguida.

4.1 Projecto Minerva

O Projecto Minerva ocorreu em Portugal entre 1985 e 1994 com a intenção de promover a introdução das novas tecnologias da informação nos ensinos básico e secundário. Este projecto surgiu dada a súbita invasão das tecnologias da informação, no contexto português, no início de 1980. Outros países, como por exemplo, a França, a Inglaterra e a Espanha tentavam também, a todo o custo, introduzir as tecnologias da informação nos seus sistemas educativos.

No geral, em toda a comunidade europeia, vivia-se um ambiente favorável à introdução das novas tecnologias na educação. No caso particular de Portugal, era visível por todo o país o surgir de inúmeras propostas de projectos neste sentido. O próprio Gabinete de Estudos e Planeamento do Ministério da Educação (GEP) conduziu o projecto DC/3 que procurava estudar o impacto do uso das novas tecnologias da informação no ensino básico. Beneficiando deste período político propício que se vivia em Portugal, o projecto Minerva arranca em três grandes períodos. Num primeiro momento, designado por “fase-piloto” que decorreu entre 1985 e 1988, organizou-se o surgir do projecto. Formaram-se equipas dinamizadoras, prepararam-se as infra-estruturas, identificaram-se “pontos estratégicos e operacionais do sistema de ensino”, “afinaram-se conceitos e critérios” e “analisaram-se soluções”. O segundo período teve lugar entre 1988 e 1992 e corresponde à fase operacional do projecto, com o aumento significativo do número de escolas envolvidas. O último período, entre 1992 e 1994, corresponde ao encerramento do projecto, fase dedicada ao balanço e à definição de novos rumos para a continuação da promoção da introdução das tecnologias da informação no sistema educativo português. Este foi, atendendo à sua dimensão, um dos projectos mais marcantes na sensibilização de professores e alunos para as novas tecnologias. Outros projectos de menor dimensão foram implementados em simultâneo com o projecto Minerva, nomeadamente os projectos IVA (Informática para a Vida Activa) e Forja (Fornecimento de equipamentos, suportes lógicos e acções de formação para professores). O projecto IVA foi lançado entre 1990 e 1992, permitindo a criação e apetrechamento de laboratórios de informática em vinte e oito escolas secundárias e incentivar a cooperação entre as escolas e as entidades locais para a concretização de projectos informáticos. À semelhança do projecto IVA, o projecto Forja que teve lugar em 1993, pretendia equipar as escolas com hardware e software, bem como a formação de professores. Porém, este último projecto não abrangeu tantas escolas quantas as pretendidas devido à indisponibilidade financeira do Ministério da Educação. Para mais informações acerca da formação, execução e balanço do projecto Minerva, consultar [7].

4.2 Edutic

Depois da conclusão do projecto Minerva, em 1995, é introduzido um novo Programa das Tecnologias de Informação e Comunicação na Educação - o Edutic. O Edutic era composto por 6 subprogramas cujos objectivos essenciais eram o apoio à produção e edição de software educativo, a formação de professores, a criação de centros de excelência que acompanhassem o desenvolvimento e o financiamento de projectos educativos das escolas, a investigação tecnológica em diferentes vertentes, a disseminação da informação de interesse para a educação na Internet e a promoção da cooperação internacional, em especial com países da União Europeia, com os PALOP e a RIBIE (rede ibero-americana de informação na educação). No entanto, a concretização de grande parte destes subprogramas não foi possível devido à falta de aprovação dos seus regulamentos (ver [17]).

4.3 Programa Nónio - séc XXI

A 4 de Outubro de 1996, pelo despacho n. 232/ME/96, do Sr. Ministro da Educação, é criado o programa Nónio-Século XXI, uma homenagem ao grande matemático, geógrafo e pedagogo português, Pedro Nunes (1502-1578). Este programa vai assumir basicamente os mesmos objectivos do EDUTIC, como por exemplo, o apetrechamento das escolas com equipamentos multimédia e o acompanhamento na formação inicial e contínua dos docentes; o apoio no desenvolvimento de projectos por parte das escolas; o incentivo à criação de software educativo, entre outras que podem ser consultadas de forma detalhada em www.giase.min-edu.pt/nonio.

4.4 CRIE (Computadores, Redes e Internet na Escola)

O equipa do CRIE foi criada em 1 de Julho de 2005 por Despacho do Ministério da Educação. Esta equipa tem como objectivo principal agregar as diferentes iniciativas numa única no Ministério da Educação e tem como propósito dinamizar a integração das TIC na educação. Para tal actua em três grandes vertentes consideradas no despacho: currículo e formação de professores; dinamização da utilização das TIC e equipamento; apetrechamento e manutenção.

Defendem que os alunos têm que saber trabalhar com as TIC, desenvolver actividades práticas de modo a poderem usá-las, de um modo transversal, em todas as disciplinas como uma ferramenta de aprendizagem e não apenas nas disciplinas directamente ligadas à informática. Procuram trabalhar no sentido de provocar alguma mudança das práticas tradicionais de ensino, [33].

Com estes e outros projectos de menor escala, desenvolvidos muitas vezes localmente em cada escola, Portugal conta hoje, segundo dados de 2006 da European Commission Information Society and Media, no seu estudo intitulado “Information and Communications Technologies (ICTs) in Schools”, com uma média de 5.4 computadores ligados à Internet por cada 100 alunos nos diferentes tipos de escolas existentes (primárias, básicas, secundárias e profissionais). Este último tipo de escolas são as que apresentam uma maior taxa, 15.3 computadores por cada 100 alunos. Estes e outros dados relativos a outros países da Europa podem ser consultados no apêndice B.

5 Introdução das novas tecnologias nas escolas do mundo

Fazendo uma viagem rápida à volta do mundo, vamos de seguida apresentar, de forma geral, o que se passa na União Europeia; na Ásia, em particular na Coreia e no Japão; nos Estados Unidos da América e na Turquia. Nesta secção é dada grande ênfase ao caso particular da Turquia porque encontramos, no artigo [35], um estudo detalhado do que sucedeu neste país, relativamente à introdução das novas tecnologias na escola.

Iniciando a nossa viagem na União Europeia, e de acordo com os dados do estudo acima referido e cujas tabelas síntese se encontram no apêndice B, podemos destacar que os países nórdicos ultrapassam os países do sul da Europa no que diz respeito ao equipamento das suas escolas. Assim, em países do norte como por exemplo, a Noruega, o Reino Unido, a Finlândia, a Suécia e a Dinamarca, o número de computadores ligados à Internet por cada 100 alunos ronda os 20 computadores. Este valor desce substancialmente quando pensamos nos países do sul da Europa cujo valor se situa entre os 6 e os 8 computadores por cada 100 alunos.

Dando um salto até à Ásia, e tendo por base os dados presentes em [1], verificamos que na Coreia, em 2001 existiam cerca de 16 computadores por cada 100 alunos. No Japão, de acordo com a mesma fonte, 97.9% das escolas estavam equipadas com computadores ligados à Internet.

Passando pelos Estados Unidos da América, e segundo dados do estudo realizado pela European Commission Information Society and Media, em 2003 existiam, em média, 4.4 computadores por cada 100 alunos.

De uma maneira geral, as escolas encontram-se mais ou menos bem equipadas com novas tecnologias, nomeadamente com computadores. Realçamos o facto de que, em praticamente todas as referências relativas à introdução das novas tecnologias da informação e comunicação (TIC), aparecem as seguintes medidas:

- (i). equipamento das escolas com computadores ligados à Internet;
- (ii). acções de sensibilização e formação de professores;
- (iii). introdução das TIC no currículo.

A primeira medida, a menos de alguns obstáculos financeiros, é relativamente simples de implementar assim como a terceira. No caso da terceira medida, o que acontece normalmente, é a criação de uma disciplina específica. No entanto, o que se pretende é que essa utilização das TIC seja transdisciplinar, o que na maior parte das vezes não acontece. Surpreendentemente, é muitas vezes a falha na implementação da segunda medida, a razão mais apontada para o insucesso do ensino assistido por computador.

No caso particular da Turquia foram várias as iniciativas tomadas pelo ministro da educação entre 1984 e 1994, com vista à promoção da literacia do computador e do uso dos computadores nas escolas. Porém, a avaliação destas iniciativas indica que o uso dos computadores nas escolas superiores da Turquia ainda está a dar os primeiros passos, sendo apontadas as seguintes razões: falta de recursos financeiros, ausência de uma planificação adequada e a falta de software apropriado. O projecto do ministro da educação turco pretendia:

- preparar a integração dos computadores no currículo;

- criar software adequado;
- formar professores;
- adquirir hardware;
- incentivar a produção local de hardware.

Durante o ano lectivo 1985 - 1986, 1111 computadores foram comprados para equipar 101 escolas superiores, dez para os alunos e um para os professores. Dois professores de cada escola foram seleccionados para fazer formação em computadores durante cinco semanas. No ano lectivo 1987 - 1988, introduziram nas escolas um curso de informática. No ano seguinte, 850 PCs foram comprados para treinar alunos no que concerne à manutenção de hardware.

Como aspectos positivos do projecto de EAC, os turcos salientam os seguintes aspectos:

- influência positiva na motivação para a aprendizagem dos alunos;
- aumento da popularidade do conceito de EAC;
- incremento considerável no interesse dos adultos pelos computadores.

Para aspectos negativos, referem:

- o software não foi desenvolvido de acordo com o currículo em vigor;
- não houve um grande envolvimento por parte dos professores no EAC;
- os professores seleccionados não foram correctamente formados;
- as firmas privadas não desenvolveram hardware e software adequados;
- o envolvimento das firmas privadas ficou aquém das expectativas.

Em Maio de 1996, o Concílio Nacional de Educação focou o seu trabalho em cinco frentes, por forma a reconstruir o sistema de educação turco, de modo a fazer face ao desenvolvimento social, científico e tecnológico do século XXI. Essas cinco frentes foram:

- educação primária e sua orientação;

- revisão no sistema do ensino secundário;
- ajuste no modo de transição para o ensino superior;
- identificação das principais necessidades educacionais da sociedade turca;
- financiamento da educação.

Porém este projecto não decorreu como o esperado e em 2000, o Banco do Mundo suportou um novo projecto intitulado “Project for Globalization in Education 2000”. Este projecto, que pretendia o uso da tecnologia em cada nível de escolaridade de modo a criar uma sociedade adaptada às tecnologias da informação, foi um grande passo para o sistema educativo da Turquia. Com este projecto, 2451 escolas primárias e secundárias foram guarnecidas com laboratórios de informática em 80 vilas e 921 cidades da Turquia. Em cada uma destas escolas as salas de informática forma equipadas com computadores, impressoras, scanners, software específico e outros equipamentos tecnológicos. As empresas ofereceram um ano de acesso grátis à Internet e o privilégio dos vizinhos das escolas usufruírem de Internet grátis ao fim-de-semana. Seguiu-se uma nova fase do projecto em mais 3000 escolas.

No entanto, apesar de todos estes projectos e dos esforços realizados desde 1984 a 2002, os turcos sentem que o uso do computador no ensino ainda se encontra numa fase embrionária do seu desenvolvimento. Mencionam a falta de uma planificação efectiva, a ausência de uma boa administração dos projectos de EAC, a desadaptação do software e hardware utilizados, a carência de apoios financeiros e a falta de formação dos professores para o EAC, como principais problemas para o pouco sucesso dos projectos implementados. Mais detalhes sobre este estudo podem ser consultados em [35].

Vejamos o que acontece no resto do mundo, iniciando a viagem na Europa. Ora, voltando ao estudo da European Comission, verificamos que, mais uma vez, os países nórdicos levam vantagem relativamente aos países do sul da Europa. De acordo com dados de 2006 desse estudo, 90% dos professores dos países nórdicos utilizaram o computador na sala de aula, contrastando com cerca de 70% dos professores dos países do sul da Europa.

Fazendo uma ponte até à Ásia, chegamos à Coreia onde 57.19% das disciplinas utilizam webboard nas suas salas de aula e onde mais de metade das

disciplinas são leccionadas recorrendo a material multimédia disponibilizado na Internet. Notamos ainda uma grande percentagem de professores, 96,10%, que utilizam o endereço electrónico e onde apenas 18.45% tem página pessoal na Internet.

Mais dados relativos à Ásia, nomeadamente, sobre a Coreia podem ser consultados no apêndice B.

6 Ensino assistido por computador(EAC)

Com estas e outras iniciativas, o computador entra no sistema educativo e surge o chamado ensino assistido por computador que pretende fazer frente às exigências do ensino dos dias de hoje.

6.1 Modos de Ensino Assistido por Computador

Existem vários tipos de EAC. Numa primeira fase utilizava-se o *ensino programado*, no qual o aluno adquiria o conhecimento através de um leque de perguntas-respostas predeterminadas com base nas respostas esperadas. O conjunto de perguntas-respostas é apresentado de forma sequencial no qual o aluno avança caso as respostas sejam correctas ou recua se responder incorrectamente. Desta forma pretende-se adaptar o ensino aos diferentes ritmos de aprendizagem dos alunos. No entanto, entre 1960 e 1970, falharam todas as tentativas de implementar este tipo de ensino nas escolas. Posteriormente, e após algum tempo de reflexão, concluiu-se que este tipo de ensino apenas é vantajoso em casos particulares. Outra forma de EAC são os *exercícios repetitivos*. Neste caso, o computador oferece ao aluno exercícios e problemas para os quais ele tem que encontrar solução. Se acertar, o computador apresenta exercícios e problemas de dificuldade superior; se errar, o computador diminui o grau de dificuldade, podendo até sugerir ao aluno que solicite a ajuda do professor. Com isto, tenciona-se orientar e estimular o aluno no processo de aquisição e consolidação de conceitos básicos e essenciais. Os exercícios repetitivos são essencialmente utilizados fora da sala de aula. Para atenuar a frieza da relação computador-aluno, apareceu o *modo diálogo*. Neste tipo de EAC todo o tipo de pergunta ou resposta do aluno leva o sistema a desencadear

várias reacções: aconselhar outro modo de resolução; levar o aluno a rever os conceitos básicos ou fazê-lo descobrir o que está errado no seu raciocínio. Este método exige uma grande e cuidadosa preparação do material a apresentar. Ao contrário dos dois métodos anteriores, este desenvolve de forma mais intensa as capacidades intelectuais dos alunos. Existe também a *simulação* particularmente utilizada em ciências experimentais como por exemplo, a física. A simulação é também utilizada a nível militar, nomeadamente na força aérea portuguesa. Segundo palavras do Major Joaquim Lima [12], “a simulação desempenha um papel importante no treino, não só por bem fundadas razões relacionadas com o treino, mas também por razões de custo, segurança e protecção ambiental.” Outra tendência mais actual é a *construção de programas de computador* como meio de desenvolvimento das capacidade de formulação e resolução de problemas. O *jogos educativos* são outro método que pode ser utilizado para desenvolver aptidões intelectuais. Funcionam essencialmente como motivadores para a prática das matérias aprendidas. No entanto, é indispensável a sua escolha cuidada tendo em conta os objectivos do ensino. Por fim, os *programas de aplicação* auxiliam a resolução de problemas, deixando de parte o árduo e laborioso trabalho dos cálculos, fomentando a formulação e a generalização de problemas [13].

Apesar da implementação de todas as iniciativas e da evolução tecnológica dos últimos anos, temos consciência de que a tecnologia, concretamente o computador, ainda está bastante longe do quotidiano da escola, ficando muitas vezes destinado unicamente às salas de informática ou de estudo. Tal como diz Nicholas Negroponte, citado por Albano, “Pero a través del mundo de hoy, las prácticas educativas están tristemente retrasadas. Pese a que los avances científicos y tecnológicos han transformado radicalmente la agricultura, la medicina y la industria, el modo en que los niños aprenden no ha sufrido cambios e sigue, en su mayor parte, basado en ideas heredadas de los siglos anteriores.” [3]. Também Jaime Carvalho e Silva questiona: “Como compreender então que a escola “fuja” ao uso da tecnologia e, por exemplo, os alunos não saibam utilizar os correctores ortográficos do software dos seus computadores, façam cálculos estatísticos laboriosamente à mão (ou usem amostras de 3 ou 4 elementos para poderem fazer as contas à mão), não saibam fazer estimativas e usar criticamente a calculadora, não saibam simular numa folha

de cálculo uma situação de empréstimo, raramente desenvolvam projectos via Internet, etc., etc, etc.?” [16]. Felizmente, começa a ser visível, um pouco por todo o país, o uso da tecnologia no processo de ensino-aprendizagem em todos os níveis de ensino.

Na nossa opinião, o Projecto Matemática Ensino (PmatE) da Universidade de Aveiro, é pioneiro na dinamização do uso do computador no ensino da matemática. Actualmente abrange todos os graus de ensino desde o 1º ciclo do ensino básico até ao ensino superior, como veremos no capítulo 4.

6.2 Porquê o Ensino Assistido por Computador

A aplicação dos computadores na educação passa muito pela utilização de software específico para as diferentes disciplinas. Assim sendo, o cuidado na selecção desse software, tendo em conta o tipo de interacção que estabelece com o aluno, a ênfase no ensino ou na descoberta, entre outros aspectos, é fundamental. Ciente deste facto, a SPA - Software Publishers Association, actualmente denominada por Software & Information Industry Association, tem publicado relatórios sobre o efeito do uso das tecnologias na escola. Em Agosto de 2000 surge o “2000 Report on the Effectiveness of Technology in Schools” [4]. Este relatório faz referência aos resultados obtidos em 311 projectos de investigação, dos quais 135 foram publicados em revistas especializadas e 56 são teses de doutoramento. A maior parte dos estudos foram desenvolvidos nos Estados Unidos entre o final dos anos 1980 e o final do século XX e mostram que o uso da tecnologia como ferramenta de aprendizagem tem um significativo efeito positivo no sucesso dos alunos, na sua atitude e na sua auto-estima, e na sua interacção com os professores e com os colegas. As conclusões deste relatório adaptam-se a alunos de todas as idades, desde o pré-escolar à universidade, sendo os resultados mais visíveis entre os alunos com baixo índice de sucesso escolar anterior, ou com necessidades educativas especiais.

O relatório anterior, “Report on the Effectiveness of Technology in Schools 1990-97”, citava que a eficiência do uso da tecnologia é influenciada pela população estudantil, pelo software utilizado, pela actividade do professor e pelo tipo de acesso à tecnologia que é facultado. Segundo estes dois relatórios, os Estados Unidos experimentaram, durante os anos 1990, um crescimento

lancinante no uso das tecnologias educativas baseadas nos computadores. Estatísticas referem que o número de computadores instalados em escolas americanas K-12 ¹ cresceu 208% entre os anos lectivos 1989-90 e 1995-96. Os estudos de investigação presentes nestes relatórios analisam metodologias variadas, como por exemplo, comparam métodos pedagógicos que incluem o uso da tecnologia com métodos tradicionais; estudam diferentes concepções de software ou o uso de tecnologia em diferentes ambientes de aprendizagem. Apesar de toda a diversidade dos estudos, estes chegam praticamente todos às mesmas conclusões das quais destacamos as seguintes:

- O uso da tecnologia tem um significativo efeito positivo no sucesso dos alunos, sucesso esse revelado em todas as disciplinas e entre alunos das diferentes faixas etárias, mesmo aqueles que necessitam de atenção especial;
- A tecnologia tem um efeito positivo na postura dos alunos relativamente à aprendizagem. A sua utilização motiva o aluno para a aprendizagem, aumenta a auto-confiança e auto-estima dos alunos, principalmente quando o aluno assume o comando do seu processo de aprendizagem;
- A eficácia do uso da tecnologia na educação é influenciada pelas características específicas da população estudantil, pelo tipo de software utilizado, a forma como o professor encara o seu papel, o modo como os alunos se encontram agrupados e o nível de acesso e manipulação da tecnologia.

Mais detalhes acerca dos estudos realizados podem ser encontrados no próprio relatório [4] ou no artigo [17].

Outro estudo [35], desta vez levado a cabo na Turquia, comparou os projectos desenvolvidos em EAC no mundo e concluiu que:

- EAC é mais benéfico para estudantes novos do que para estudantes mais velhos;
- EAC é mais proveitoso para alunos de estratos socio-económicos mais baixos do que para estudantes economicamente favorecidos;

¹Desde o pré-escolar até ao 12º ano de escolaridade

- Os alunos com baixas expectativas escolares beneficiam mais do EAC do que os alunos com altas expectativas escolares;
- EAC é mais eficiente no ensino de assuntos de baixo teor cognitivo do que temas com alto teor cognitivo;
- Os alunos com necessidades educativas especiais conseguem chegar mais longe com a ajuda do EAC do que apenas com o ensino tradicional;
- Não se verificam diferenças significativas na eficiência do EAC entre alunos masculinos e alunos femininos;
- O uso do EAC como complemento ao ensino tradicional, produz melhores resultados do que o uso exclusivo dos métodos tradicionais;
- Os estudos são inconclusivos quando comparam o uso absoluto do método tradicional com o uso único do EAC;
- Os alunos que utilizam os processadores de texto escrevem com mais qualidade do que os restantes;
- Os alunos aprendem mais rapidamente com o EAC do que com métodos tradicionais;
- Os alunos retêm mais o que aprendem com o EAC do que o que aprendem usando métodos exclusivamente convencionais.

Capítulo 4

Projecto Matemática Ensino (PmatE)

“Pelas minhas contas, projecto no espaço e no tempo um valor infinito de trabalho a prosseguir pelo Projecto Matemática Ensino para o desenvolvimento integral do sistema e geometria do Ensino da Matemática no universo português.”

A Reitora da Universidade de Aveiro

Prof. Doutora Maria Helena Nazaré

1 O projecto

O Projecto Matemática Ensino (PmatE) da Universidade de Aveiro tem desenvolvido desde 1990 uma plataforma de ensino assistido (PeA). Este projecto surgiu, e segundo palavras do fundador, o Professor Doutor David Vieira do Departamento de Matemática da Universidade de Aveiro, de “um desafio a dois colegas mais novos - António Batel Anjo e Maria Paula Carvalho - para criarmos um sistema informático de avaliação em disciplinas do ensino superior com grande número de alunos”. Devido à insuficiência de meios naquele tempo, decidiram experimentar o seu sistema informático em alunos dos 7º ano de escolaridade de uma escola de Aveiro. Atendendo ao sucesso que obtiveram com esta boa experiência, decidiram criar a competição matemática EQUA-mat, da qual tanto se ouve falar. Mas não pararam por aqui, o entusiasmo de

alunos e professores com esta competição, encorajou a equipa do projecto em continuar a desenvolver cada vez mais o seu sistema informático e ampliar o sistema de diversos modos. A nível de competições contam já com o EQUA-mat (3º Ciclo), mat12 (secundário), MINImat (1º ciclo) e MAISmat (2º ciclo). É de realçar ainda a contribuição da plataforma do PmatE a nível do Ensino Superior. Nos últimos anos esta plataforma tem sido utilizada para efeitos de avaliação em algumas disciplinas da Universidade de Aveiro, nomeadamente, Cálculo I, Cálculo II, Cálculo III, Sistemas Digitais, entre outras. Actualmente esta plataforma encontra-se apenas disponível na Internet, abrangendo, como já foi referido, vários graus de ensino, desde o básico até ao superior. Fazemos notar ainda a existência de outras iniciativas do projecto com o intento de promover o gosto pela matemática junto dos mais novos e apoiar o ensino desta disciplina em Portugal e no estrangeiro. Mais informação acerca destas e outras iniciativas podem ser consultadas no endereço electrónico *pmate.ua.pt*.

Na secção 2 damos a conhecer o principal produto do PmatE, os modelos geradores de questões e na secção 3 apresentamos os resultados da experiência de ensino da disciplina de Cálculo III da Universidade de Aveiro no primeiro semestre do ano lectivo 2005/2006 com o apoio do PmatE.

2 Modelos Geradores de Questões (MGQ)

O trabalho principal do PmatE é o desenvolvimento de software, MGQ, essencialmente para matemática. Os MGQ, [15], são criados por trabalhadores efectivos do projecto e também por professores dos diferentes níveis de ensino, nas suas componentes científica, pedagógica e tecnológica. Cada MGQ é constituído por um texto principal e por, pelo menos, quatro questões parametrizadas de verdadeiro/falso, podendo conter ou não imagens. Após a criação do modelo este é classificado, quer do ponto de vista científico-pedagógico, quer do ponto de vista da sua dificuldade. Com efeito, cada MGQ é etiquetado por um chamado *objectivo secundário* que corresponde ao tema que se encontra abordado no MGQ. Os conteúdos e respectivos objectivos de todas as áreas e graus de ensino abrangidos pelo PmatE encontram-se organizados na denominada *árvore de objectivos*. A Figura 4.1 ilustra a organização em árvore de uma parte dos conteúdos programáticos da disciplina de Cálculo III, cuja

- ST(291) Integração
 - OP(827) Integrais duplos
 - OS(2352) Definição
 - OM_1(5395) Definição
 - OS(2353) Propriedades
 - OM_1(5396) Propriedades
 - OS(2354) Mudança do integral duplo para o integral simples
 - OM_1(5397) Teorema de Fubini
 - OM_1(5398) Regiões verticalmente e horizontalmente
 - OS(2355) Mudança de variável
 - OM_1(5399) Coordenadas polares
 - OM_1(5400) Mudança da variável
 - OS(2356) Aplicações
 - OM_1(5401) Cálculo de áreas
 - OM_1(5402) Cálculo de volumes
 - OM_1(5403) Centro de massa de uma lâmina fina
 - OS(2399) Integrais duplos
 - OM_1(5553) Integrais duplos
 - OP(828) Integrais triplos

Figura 4.1: Excerto da Árvore de Objectivos

experiência no PmatE irá ser relatada na secção 3.

Quanto à dificuldade, cada MGQ é classificado numa escala de 1 (simples) a 5 (muito difícil). Após esta fase, o modelo é enviado para programação onde lhe é atribuído um número, número esse que vais ser usado para o identificar. Segue-se a fase de avaliação na qual a pessoa que elaborou o modelo verifica se tudo está bem. Terminada a avaliação, o modelo está pronto para ser utilizado. Uma das maiores vantagens destes MGQ é o facto de se obter diferentes concretizações do mesmo modelo. As Figuras 4.2, 4.3, 4.4 e 4.5, mostram diferentes concretizações correspondendo, cada par, ao mesmo MGQ, cujo tex está incluído no apêndice C.

3 A Experiência de Cálculo III

Dada a elevada taxa de reprovação à disciplina de cálculo III, a equipa de docentes decidiu usufruir da plataforma de ensino assistido desenvolvida pelo PmatE para enriquecer o processo de ensino-aprendizagem e avaliação desta disciplina. Assim, em Março de 2005 teve início o processo de elaboração,

Seja $f: D_f \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}^3$ uma função vectorial definida por $f(x, y, z) = (f_1(x, y, z), f_2(x, y, z), f_3(x, y, z))$.

Se f é contínua em $(a, b, c) \in D_f$ então $f(x_n, y_n, z_n)$ é uma sucessão de pontos de \mathbb{R}^3 que converge para $f(a, b, c)$, para qualquer sucessão (x_n, y_n, z_n) de pontos de D_f convergentes para (a, b, c) . C V
C F

Dizemos que f é contínua no ponto $(a, b, c) \in D_f$ se existem sucessões $(x_n), (y_n), (z_n)$ de pontos de D_f tais que $(x_n, y_n, z_n) \rightarrow (a, b, c)$ e $f(x_n, y_n, z_n) \rightarrow f(a, b, c)$. C V
C F

A função $f = f(x, y, z)$ pode não ser contínua em $(a, b, c) \in D_f$ mesmo que todas as suas funções coordenadas sejam contínuas em (a, b, c) . C V
C F

Se $(a, b, c) \in D_f$, $\lim_{(x, y, z) \rightarrow (a, b, c)} f_1(x, y, z) = r$, $\lim_{(x, y, z) \rightarrow (a, b, c)} f_2(x, y, z) = s$, $\lim_{(x, y, z) \rightarrow (a, b, c)} f_3(x, y, z) = t$ e $f(a, b, c) = (r, s, t)$ então f é contínua em (a, b, c) . C V
C F

Figura 4.2: Concretização de um MGQ sobre continuidade de funções vectoriais

programação e avaliação de modelos que abarcavam todos os conteúdos programáticos de cálculo III. Os modelos, de carácter maioritariamente teórico, destinavam-se a ser mais uma ferramenta de estudo para os alunos. Deste modo, foram disponibilizados ao longo do primeiro semestre do ano lectivo 2005/2006 e em conformidade com os conteúdos leccionados, alguns modelos de modo que os alunos os utilizassem como ferramenta de estudo com vista à preparação dos testes escritos da avaliação contínua.

Nesta primeira experiência, os modelos não foram usados como meio de avaliação dos alunos, mas como ferramenta de selecção de alunos para a realização dos testes escritos. Para cada um dos três testes os alunos tinham, previamente, de obter um chamado “passaporte” na PeA. O passaporte consistia numa prova de avaliação construída com modelos relativos aos assuntos a serem avaliados no teste escrito. Para poderem realizar o teste escrito, os alunos teriam de alcançar uma classificação igual ou superior a 16 valores em, pelo menos uma das 15 tentativas disponíveis para cada um dos referidos pas-

Seja $f: D_f \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}^2$ uma função vectorial definida por $f(x, y, z) = (f_1(x, y, z), f_2(x, y, z))$.

Dizemos que f é contínua no ponto $(a, b, c) \in D_f$ se existem sucessões $(x_n), (y_n), (z_n)$ de pontos de D_f tais que $(x_n, y_n, z_n) \rightarrow (a, b, c)$ e $f(x_n, y_n, z_n) \rightarrow f(a, b, c)$.

☐ V
☐ F

A função $f = f(x, y, z)$ é contínua em $(a, b, c) \in D_f$ se e só se as suas funções coordenadas $f_1(x, y, z), f_2(x, y, z)$ são contínuas em (a, b, c) .

☐ V
☐ F

Se f é contínua em $(a, b, c) \in D_f$ então $f(x_n, y_n, z_n)$ é uma sucessão de pontos de \mathbb{R}^3 que converge para $f(a, b, c)$, para qualquer sucessão (x_n, y_n, z_n) de pontos de D_f convergentes para (a, b, c) .

☐ V
☐ F

Se f é contínua em $(a, b, c) \in D_f$ e $f(a, b, c) = (r, s)$ então $\lim_{(x, y, z) \rightarrow (a, b, c)} f_1(x, y, z) = r$,
 $\lim_{(x, y, z) \rightarrow (a, b, c)} f_2(x, y, z) = s$.

☐ V
☐ F

Figura 4.3: Segunda concretização de um MGQ sobre continuidade de funções vectoriais

saportes. Salientamos que no último passaporte houve a necessidade de se acrescentar mais 10 tentativas devido a problemas no servidor do PmatE que dificultavam o acesso, por parte dos alunos, às tentativas. Os resultados que se seguem referem-se a cada um dos três momentos de avaliação que os alunos que optaram pela avaliação contínua tiveram de se sujeitar.

O gráfico 4.6 mostra o número total de provas realizadas em cada um dos três passaportes por 956, 785 e 633 alunos, respectivamente. O número de alunos foi decrescendo ao longo dos três passaportes dado que os alunos que não obtiveram o primeiro passaporte não podiam tentar nem o segundo nem o terceiro passaporte e os alunos que não conseguiram atingir o segundo passaporte estavam impossibilitados de tentar alcançar o terceiro.

Verificamos a diminuição do número de provas do primeiro para o segundo passaporte e um ligeiro aumento do número de provas do segundo para o terceiro passaporte. A diminuição deve-se ao facto de que os alunos que não atingiram o primeiro passaporte não podiam tentar obter o segundo nem o ter-

[1147]

Sejam C_1 e C_2 as curvas de equação $\frac{1}{4}x^2 + \frac{1}{16}(y-4)^2 = 1$ e $\frac{1}{25}x^2 + \frac{1}{25}(y-4)^2 = 1$, com orientação positiva. D a região limitada pelas curvas C_1 e C_2 .

Seja $F(x, y) = (-y, y^2)$. A área da região D é a soma do trabalho realizado pelo campo F ao longo da curva C_2 com o trabalho realizado pelo campo F ao longo da curva C_1 .

☐ V
☐ F

Seja $F(x, y) = (y^2, x)$. Então, $\oint_{C_2} F(x, y) d\vec{r} - \oint_{C_1} F(x, y) d\vec{r} \neq \iint_D 1 - 2y \, dx dy dx dy$.

☐ V
☐ F

As funções: $r_1(t) = (2\cos t, -4\sin t + 4)$ $t \in [0; 2\pi]$ e $r_2(t) = (5\cos t, -5\sin t + 4)$ $t \in [0; 2\pi]$ são parametrizações regulares, para as curvas C_1 e C_2 , respectivamente, que respeitam a orientação.

☐ V
☐ F

A região D não é simplesmente conexa.

☐ V
☐ F

Figura 4.4: Concretização de um MGQ sobre o Teorema de Green

ceiro, uma vez que foram automaticamente eliminados da avaliação contínua. O ligeiro aumento do número de provas do segundo para o terceiro passaporte é consequência do acréscimo do número de tentativas explicado anteriormente.

No gráfico 4.7 estão presentes os resultados comparativos entre o número de alunos que acederam às provas e o número de alunos que, efectivamente, alcançaram o passaporte, isto é, que obtiveram uma classificação igual ou superior a 16 valores em, pelo menos, uma das tentativas disponíveis para cada um dos três passaportes. Tendo em conta os dados recolhidos, podemos afirmar que cerca de 80 por cento dos alunos que acederam às provas conseguiram obter passaporte, ou seja, conseguiram realizar o respectivo teste escrito.

Fazendo o balanço da implementação desta nova metodologia de ensino, e comparando os resultados com os obtidos no ano lectivo anterior, 2004/2005, em que a avaliação consistia apenas num exame final, verificamos que o número de avaliados passou de 599 para 1000 e o número de aprovações de 195 para 491. Desta forma, a taxa de aprovados por inscritos aumentou de 14,0% para

[1147]

Sejam C_1 e C_2 as curvas de equação
 $x^2 + \frac{1}{3}(y-2)^2 = 1$ e $\frac{1}{25}x^2 + \frac{1}{9}(y-2)^2 = 1$, com orientação positiva.
 D a região limitada pelas curvas C_1 e C_2 .

Seja $F(x, y) = (2y, 4x)$ A área da região D é a diferença entre o trabalho realizado pelo campo F ao longo da curva C_2 e o trabalho realizado pelo campo F ao longo da curva C_1 .	<input type="radio"/> V <input type="radio"/> F
O campo vectorial F definido por $F(x, y) = (6x^5y, x^2y)$ não é conservativo.	<input type="radio"/> V <input type="radio"/> F
A região D não é conexa.	<input type="radio"/> V <input type="radio"/> F
A área da região D é o valor de $\frac{1}{5} \oint_{C_2} -5ydx + y^5 dy - \frac{1}{5} \oint_{C_1} -5ydx + y^5 dy$	<input type="radio"/> V <input type="radio"/> F

Figura 4.5: Segunda concretização de um MGQ sobre o Teorema de Green

35,7% e a taxa de aprovados por avaliados subiu de 32,6% para 49,1%. Estes dados foram retirados do Relatório da disciplina de Cálculo III, 2003-2006, elaborado na Universidade de Aveiro pelo regente da cadeira.

Atendendo à melhoria dos resultados obtidos com esta experiência, a equipa de docentes da disciplina de Cálculo III decidiu, no presente ano lectivo, 2006/2007, realizar quatro testes informatizados na plataforma do PmatE com peso na nota final dos alunos.

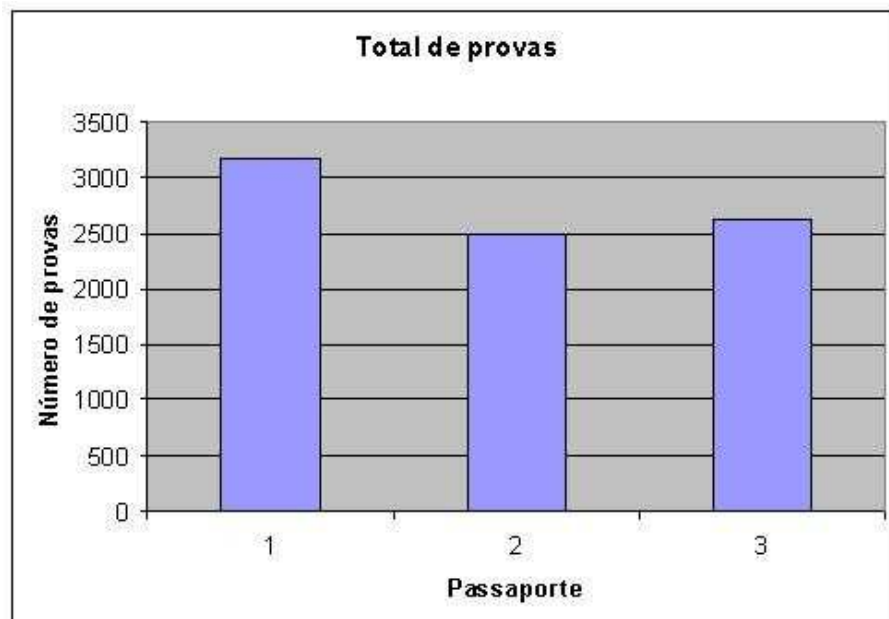


Figura 4.6: Número total de provas realizadas

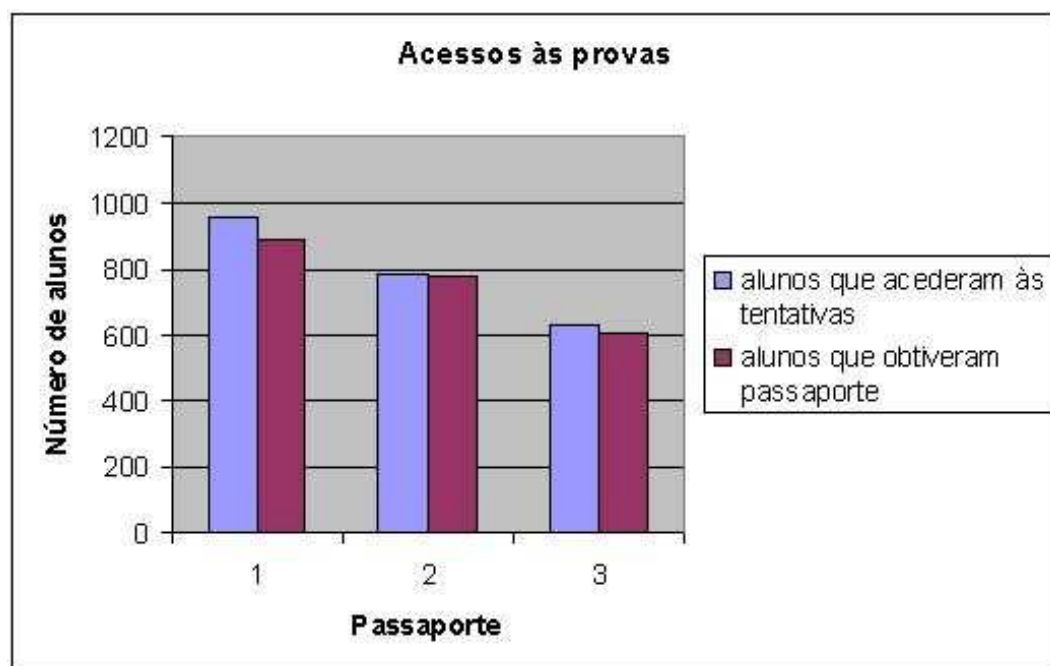


Figura 4.7: Número total de acessos

Capítulo 5

Sistema de Ensino Assistido por Computador

“Diz-me e eu esquecerei, ensina-me e eu lembrar-me-ei, envolve-me e eu aprenderei.” (Provérbio Chinês)

Tal como vimos no capítulo 3, o uso dos computadores na educação iniciou-se, de forma mais notável, a partir de 1980. No entanto, a maior parte dos sistemas de ensino assistido por computador utilizados não tinha em conta as necessidades e dificuldades individuais de cada aluno. Este tipo de sistemas reage de igual forma para com todos os alunos de maneira semelhante à seguinte: se a resposta 21 estiver correcta então o aluno deve responder à questão 54, caso contrário, responde à questão 32, [25]. Como se pode verificar, as capacidades individuais de cada aluno são desprezadas e o sistema tem apenas em consideração se a resposta está correcta ou errada. Para colmatar esta falha, vários investigadores pensaram na possibilidade de introduzir uma espécie de assistente pessoal para cada aluno e que estaria atento às suas dificuldades. Em 1984, Benjamin Bloom (21 de Fevereiro de 1913 - 13 de Setembro de 1999), psicólogo educacional com grande contribuição na classificação de objectivos da educação e na teoria do “master learning”, mostrou que estudantes que receberam ensino individualizado tinham um maior e melhor desempenho do que alunos ensinados em turmas, [23]. É claro que nos dias de hoje é impensável ter um professor para cada aluno. Temos portanto que tirar partido das novas tecnologias e surge assim o que se denomina por

Ensino Assistido por Computador Inteligente e desenvolvem-se Sistemas de Ensino Assistido por Computador Inteligentes (SEACI). Esta nova forma de ensino assistido por computador tem sido acompanhada desde à vários anos por investigadores de diversas áreas, nomeadamente, educação, psicologia e computação.

1 Sistemas Inteligentes

Os SEACIs podem ser vistos como um conjunto de diferentes componentes interligadas. Tendo em conta a divisão de Woolf, podemos dizer que os SEACIs são constituídos por quatro grandes componentes: a componente do aluno, a componente pedagógica, a componente do domínio do conhecimento e a componente da comunicação. Porém, Joseph Beck, Mia Stern e Erik Haugsjaa, identificaram uma quinta parte, à qual chamaram “expert model” a qual, segundo estes autores, fazia parte da componente do domínio do conhecimento na divisão de Woolf. A Figura 5.1 fornece uma visão esquemática da articulação das diferentes partes constituintes de um SEACI, segundo a divisão de Joseph Beck, Mia Stern e Erik Haugsjaa. Vejamos qual a função de cada uma delas em particular, [25].

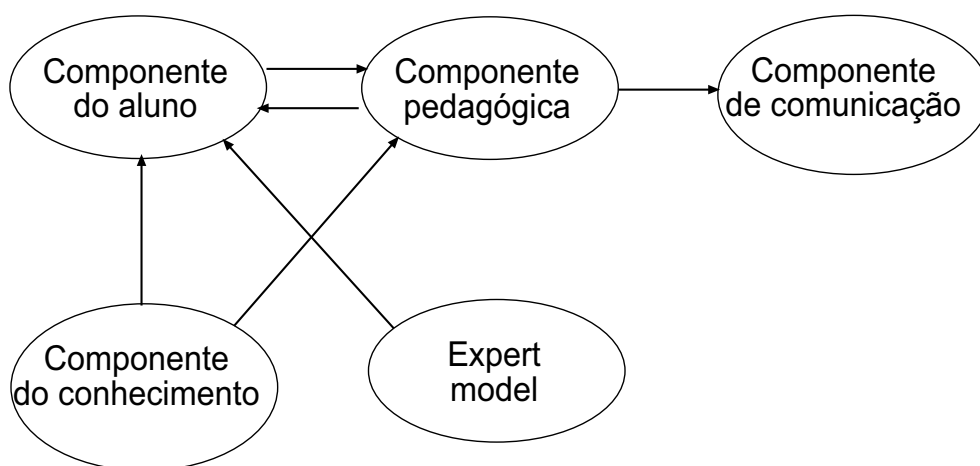


Figura 5.1: Articulação das diferentes componentes de um Sistema Inteligente

A componente do aluno contém a informação específica de cada aluno, isto é, quais os conhecimentos e as capacidades de cada um. Por exemplo, Dr.

Valerie Shute apresentou o seguinte simples exemplo hipotético de um sistema de ensino de aritmética. Suponhamos que três alunos, Jack, Jill e Bill são confrontados com duas adições e respondem como mostra a figura 5.2, [22].

$\begin{array}{r} 22 \\ + 39 \\ \hline (5)1 \end{array}$	$\begin{array}{r} \overset{1}{2} \overset{1}{2} \\ + 39 \\ \hline (1)61 \end{array}$	$\begin{array}{r} \overset{1}{2} \overset{1}{2} \\ + 39 \\ \hline 6(2) \end{array}$
$\begin{array}{r} 46 \\ + 37 \\ \hline (7)3 \end{array}$	$\begin{array}{r} \overset{1}{4} \overset{1}{6} \\ + 37 \\ \hline (1)83 \end{array}$	$\begin{array}{r} \overset{1}{4} \overset{1}{6} \\ + 37 \\ \hline 8(5) \end{array}$
Jack	Jill	Bill

Figura 5.2: Diagnóstico dos diferentes tipos de erros matemáticos nas respostas dos 3 alunos testados

Analisando as suas respostas, observamos que os três respondem incorretamente mas cometem erros diferentes. O aluno Jack esquece-se do que vai de trás; o aluno Jill, ao contrário do Jack, conta mais do que uma vez o que vai de trás; o Bill erra logo na adição de dois algarismos. Outro exemplo, mas desta vez num domínio mais complexo, é o Tactical Action Officer (TAO) SEACI, desenvolvido por Stottler Henke para a Marinha dos Estados Unidos, cujo objectivo é ensinar as regras tácticas em cenários realistas. A Figura 5.3, retirada do artigo [22], ilustra uma realização deste sistema de ensino inteligente.

Na parte esquerda da janela, o estudante comanda o navio numa situação de guerra simulada. No final de cada prestação do aluno é apresentada a lista de acções realizadas pelo aluno, como se encontra na zona mais à direita da figura, em que as acções com bolinha vermelha são incorrectas e as com bolinha verde são as correctas. Este tipo de informação é guardada na componente do aluno para posterior utilização.

Na componente pedagógica está a informação sobre as decisões que devem ser tomadas pelo sistema. Por exemplo, quando é que deve rever um ou outro conteúdo ou quando apresentar um novo tema e qual o tema a apresentar. Es-

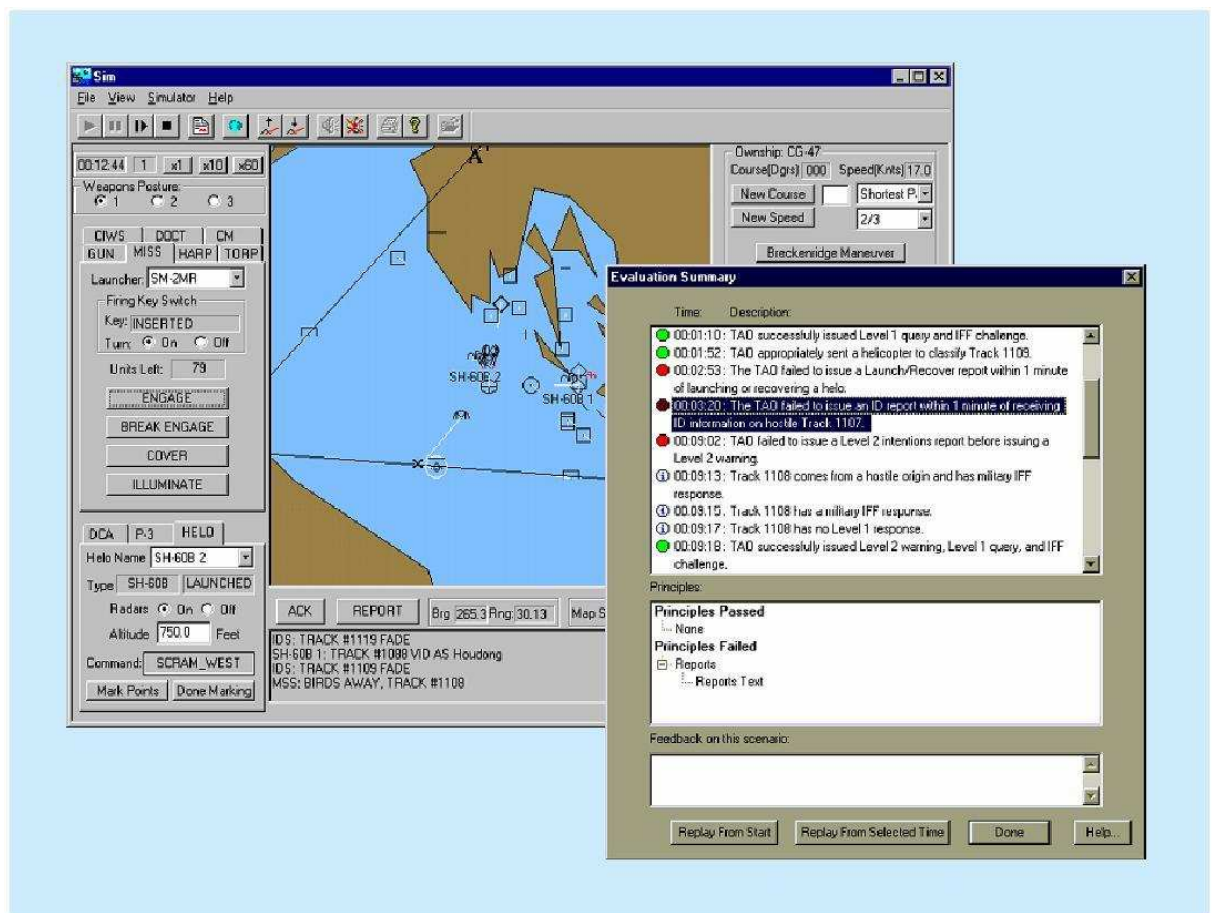


Figura 5.3: Realização do SEACI TAO e relatório das tarefas realizadas

tas decisões são tomadas de acordo com a informação guardada na componente anterior.

A componente do domínio do conhecimento é responsável por reter os assuntos que irão ser apresentados aos alunos. Existem vários estudos que se debruçam sobre o modo como tal informação deve ser armazenada.

A componente da comunicação é a parte do sistema que interage directamente com o aluno. Encontrar a maneira mais eficaz e agradável de apresentação dos conteúdos aos alunos tem sido também alvo de investigações.

Finalmente temos o “expert model”. Podemos afirmar que o “expert model” é a camada do sistema que guarda as respostas de um aluno virtual inteligente às mesmas questões apresentadas ao aluno real, com as quais o

sistema pode comparar as respostas e identificar os conteúdos nos quais o aluno sente mais dificuldades.

Baseado na prestação do aluno, os SEACI, ao contrário dos outros SEAC, devolvem estratégias de ensino, quer a nível dos conteúdos, quer a nível de estilo, fornecendo, explicações, sugestões, exemplos, demonstrações e problemas práticos, de acordo com as necessidades diagnosticadas.

Os diferentes estudos revelam que os alunos instruídos com SEACIs aprendem, geralmente, mais rápido e têm uma maior capacidade de transmissão dos seus conhecimentos do que os outros alunos. Apresentamos de seguida dois exemplos que comprovam o que acabamos de afirmar. Na Universidade de Carnegie Mellon, em 1980, investigadores desenvolveram um SEACI, chamado LISP Tutor para o ensino de programação de computadores. Com a avaliação da implementação deste sistema, os investigadores concluíram que os alunos que utilizaram o SEACI obtiveram notas superiores, em 43%, no exame final do que os restantes alunos. Mais verificaram que os alunos do ensino tradicional necessitavam de mais 30% de tempo para a resolução de problemas complexos de programação do que os alunos que utilizaram o SEACI. Outro SEACI, intitulado Sherlock, foi criado no início do ano 1990 e pretendia treinar o pessoal da força aérea na resolução de problemas. Mais uma vez concluiu-se que os alunos treinados com o Sherlock tinham uma melhor prestação do que os do treino tradicional. Mais, após 20 horas de instrução com o Sherlock, agiam tão bem como técnicos com quatro anos de experiência profissional na área. Após os primeiros anos de implementação destes novos SEAC, seguiram-se anos de grande desenvolvimento e aplicação em outras áreas, [22]. Seguem-se alguns dos exemplos mais notáveis:

- SEACI desenvolvido por Alan M. Lesgold e colegas da Universidade de Pittsburgh treina técnicos de uma multinacional de semicondutores para a reparação de equipamento complexo;
- SEACI para o ensino de conteúdos matemáticos para o nível secundário semelhante ao criado por John Anderson e Ken Koedinger na Universidade de Carnegie Mellon foi testado numa escola superior pública de Pittsburgh, cujos alunos mostraram uma evolução notável na aptidão para resolver problemas, usar equações, tabelas e gráficos;

- O TAO SEACI usa cenários simulados e avalia automaticamente os alunos como foi explicado anteriormente. Com este sistema, os alunos recebem 10 vezes mais experiência do que através do modo tradicional. É usado quer na sala de aula quer fora em regime de auto-estudo.

2 A motivação

A experiência de ensino assistido por computador vivida com a disciplina de Cálculo III no primeiro semestre do ano lectivo 2005/2006, descrita no capítulo 4, motivou a criação deste sistema de ensino inovador, [14]. A ideia essencial é rentabilizar a plataforma de ensino assistido do PmatE, PeA, fazendo com que a sequência de modelos oferecida ao aluno não seja pré-determinada. Este novo sistema permite ao aluno responder a questões de diversos temas sem uma ordem pré-destinada. Após cada resposta do aluno um novo tema e, consequentemente, um modelo é escolhido, tendo em conta o desempenho do aluno nos modelos anteriores. Por exemplo, se o aluno não tem um bom desempenho num determinado tema, o sistema procura fazer surgir ao aluno modelos que estejam relacionados com temas que servem de base ao conteúdo no qual o aluno está a demonstrar dificuldades. Pelo contrário, se o aluno mostrar que domina um determinado tema, o sistema oferece ao aluno questões sobre temas mais avançados.

3 Armanezamento de Informação

Como já foi referido, este sistema de ensino assistido por computador tem o seu suporte matemático nos autómatos probabilísticos. Os estados desse autómato são, neste momento, os quarenta e um objectivos secundários relativos aos conteúdos programáticos da disciplina de cálculo III e que se encontram na árvore de objectivos do PmatE. Cada aresta tem a si associada uma determinada probabilidade que é sucessivamente actualizada ao longo do programa. Toda a informação necessária para a criação deste sistema de ensino assistido encontra-se arrecadada nas quatro matrizes A, B, C e D, respectivamente, permanência, bases, conhecimento e distância. A matriz B é uma matriz de zeros e uns que traduz a estrutura da árvore de objectivos (ver Figura 4.1 do capítulo 4) e

representa a dependência entre os diferentes objectivos. Esta matriz fornece a informação acerca dos objectivos que são (1) ou não (0) bases para outros. O grafo com os objectivos secundários e que expressa a organização da matriz B encontra-se na Figura 5.4. Os objectivos secundários em posição inferior são base para os que se encontram em cima se existir uma ligação entre eles.

A matriz D guarda a informação sobre a proximidade entre os diversos objectivos secundários. Quanto maior o valor do elemento da matriz, mais próximos estão os objectivos. Entende-se por objectivos secundários próximos aqueles que estão fortemente relacionados, do ponto de vista do seu conteúdo. Quanto mais afastados estão os objectivos secundários, menos relacionados estão. Esta relação entre os objectivos secundários é também visível na Figura 5.4, uma vez que os objectivos próximos têm a mesma cor. Assim, as matrizes B e D contêm informação que não se altera ao longo do programa.

No que diz respeito à matriz C esta é, na realidade, um vector com quarenta e uma posições que conserva a percentagem de conhecimento que o aluno possui em cada um dos objectivos secundários. Finalmente a matriz A, que também é um vector com quarenta e uma posições, contém o número de questões respondidas pelo aluno em cada objectivo secundário. Os vectores A e C guardam informação que é actualizada após cada interacção do aluno.

4 Funcionamento do Sistema

No início do programa todas as posições do vector C tomam o valor de 50 por cento e todas as posições do vector A são nulas. Tendo em conta estes valores e os valores constantes das matrizes B e D, o sistema cria o vector de probabilidades, P, utilizando uma fórmula que será posteriormente apresentada. Com os valores de P, o primeiro estado é escolhido e um modelo pendurado nesse objectivo secundário é aleatoriamente seleccionado e apresentado ao aluno. Realçamos que os valores iniciais do vector P são tais que o primeiro estado a ser escolhido não corresponde a um objectivo secundário cujo tema possua um elevado grau de dificuldade. O aluno deve começar o seu percurso pelos objectivos secundários correspondentes a temas simples, aumentando ou diminuindo o grau de dificuldade dos mesmos de acordo com o seu desempenho. Após o aluno ter respondido ao modelo que lhe foi apre-

sentado, validando as quatro afirmações, como verdadeiras ou falsas, o sistema guarda o número de respostas certas dadas pelo aluno. Se o aluno validou correctamente três ou quatro afirmações, o valor da correspondente posição do vector C é incrementado; caso o aluno valide correctamente apenas duas afirmações, o valor da correspondente posição do vector C não é alterado e se o aluno validar correctamente nenhuma afirmação ou apenas uma, o valor da respectiva posição do vector C é decrementado. Sempre que o aluno responda a um modelo de um determinado objectivo secundário, a referente posição do vector A é incrementada de uma unidade. Actualizados os vectores A e C, um novo vector de probabilidades é criado, um novo estado é seleccionado e um modelo é aleatoriamente escolhido, repetindo-se todo o processo anteriormente descrito até que o aluno demonstre ter algum conhecimento nos temas abordados. A Figura 5.5 ilustra, de forma esquemática, o modo de funcionamento do sistema.

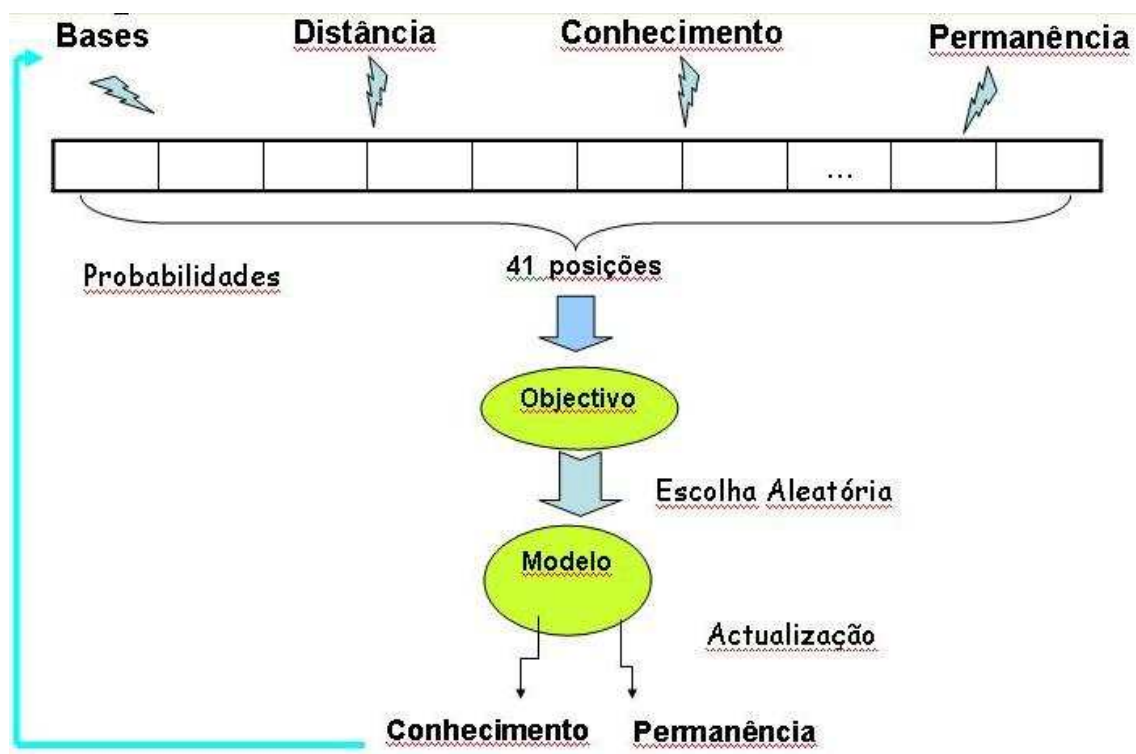


Figura 5.5: Funcionamento do Sistema

A fórmula usada para actualizar o vector das probabilidades é a seguinte

$$p_j = \frac{w_B \bar{B}_j + w_D \bar{D}_j + w_A \bar{A}_j + w_C \bar{C}_j}{w_B \sum_{i=1}^n \bar{B}_i + w_D \sum_{i=1}^n \bar{D}_i + w_A \sum_{i=1}^n \bar{A}_i + w_C \sum_{i=1}^n \bar{C}_i} \quad (j = 1 \dots n)$$

Denotamos por \bar{M} a contribuição da matriz M para o cálculo das probabilidades. Como se pode verificar, as quatro matrizes contribuem para o cálculo das probabilidades. As constantes positivas w_A , w_B , w_C e w_D medem a contribuição de cada matriz. Os somatórios em denominador garantem que

$$\sum_{j=1}^n p_j = 1,$$

como é exigido pela teoria da probabilidades.

5 Reflexões

Este simples sistema de ensino assistido por computador faz com que o aluno navegue, ao seu próprio ritmo, pelos conteúdos programáticos da disciplina de cálculo III, progredindo de conteúdos mais simples para conteúdos mais complexos.

Porém, tratando-se de um sistema em fase de desenvolvimento e de prova, alguns aspectos podem ser aperfeiçoados por forma a tornar ainda mais eficaz este sistema. Apontamos como aspectos que podem sofrer alteração a actualização do vector do conhecimento, C; a condição de paragem do sistema; o valor das constantes positivas presentes na fórmula de cálculo das probabilidades e a qualidade dos modelos. Neste momento, a actualização do vector C é feita adicionando 20 por cento à correspondente posição do vector C caso o aluno valide correctamente as 4 afirmações e em 10 por cento caso o aluno valide 3 afirmações correctamente. Se o aluno validar correctamente uma afirmação, a respectiva posição do vector C é decrementada em 10 por cento. No caso de não validar correctamente qualquer uma das 4 afirmações, a correspondente posição do vector C é decrementada em 20 por cento. Os valores destas percentagens podem ser alterados de modo a manter o aluno mais ou menos tempo no mesmo objectivo secundário. Actualmente, o aluno deixa de estar no sistema se, pelo menos, 30 das 41 posições do vector C tiver

uma percentagem superior a 90 por cento. Esta única condição de paragem pode ser modificada caso se pretenda que o aluno permaneça mais ou menos tempo no sistema. Como já referimos, as constantes positivas w_A , w_B , w_C e w_D medem a contribuição das matrizes A, B, C e D, respectivamente, no cálculo das probabilidades. Neste momento, os parâmetros estão estabelecidos no início do programa com os valores 35, 10, 45 e 10, por esta ordem. Estes parâmetros são actualizados durante a execução do programa, tomando em consideração o desempenho do aluno. Por exemplo, se o aluno demonstrar muitas dificuldades num determinado tema, o valor de w_B deve ser aumentado por forma a que o sistema encaminhe o aluno para temas que servem de base àquele no qual o aluno se encontra parado. No entanto, se o aluno mostrar ter já alguns conhecimentos, os valores de w_C e w_D devem ser incrementados por forma a levar o aluno até temas com um nível de dificuldade superior. Presentemente mais e melhores modelos estão a ser elaborados, de modo a garantir uma panóplia diversificada de modelos capazes de assegurar um processo de ensino-aprendizagem eficiente.

Trata-se de um sistema que pode perfeitamente ser adequado a outras disciplinas e a outras plataformas de ensino assistido, efectuando para isso as devidas alterações. De facto, o sistema é constituído essencialmente por duas classes, a classe aluno e a classe disciplina (ver código em A). Uma alteração do conteúdo implica apenas a alteração da classe disciplina a qual contém os dados sobre o conteúdo da PeA que é vasto, mantendo-se inalterada a classe aluno.

Bibliografia

- [1] *Indicators of information and communication thechnology use in elementary and secondary education*, ICT in the Classroom, Seventh Programming Cycle of APEID Activities, Japan (2003).
- [2] Abe and Warmuth, *On the complexity of aproximating distributions on probability automata*, Machine learning **9** (1992), 205–260.
- [3] Adilton José Pinheiro Albano, *O computador como instrumento pedagógico. estudo efectuado junto dos professores dos 2º e 3º grupos do 2º ciclo do ensino básico do centro da área educativa de castelo branco*, Universidade da Beira Interior, 2002.
- [4] Software & Information Association, *2000 report on the effectivenesss of technology in schools*, Washington, DC: Software and Information Industry Association, 2000.
- [5] Taylor L. Booth, *Sequential machines and automata theory*, John Wiley and Sons, Inc., 1967.
- [6] S.C. Coutinho, *Operações com autómatos finitos*, Notas de Aula (2005).
- [7] João Pedro da Ponte, *O projecto minerva, introduzindo as nti na educação em portugal*, Lisboa: DEPGEF Ministério da Educação, 1994.
- [8] ———, *As novas tecnologias e a educação*, Texto Editora, Lda, 1997.
- [9] Luiz Eduardo Soares de Oliveira e Marisa Emika Morita, *Introdução aos modelos escondidos de markov*, (1998).
- [10] Rodrigo Nonamor Pereira Mariano de Souza, *Autómatos e algoritmos de busca de padrões*, (2000).

- [11] Radu Marculescu e Massoud Pedram Diana Marculescu, *Stochastic sequential machine synthesis constrained sequence generation*, Proceedings of the 33rd Annual Conference on Design Automation (1996).
- [12] Joaquim Manuel Martins do Vale Lima, *As novas tecnologias no ensino*, Air & Space Power Journal **2** (2006).
- [13] Teresa Mendes e A. Dias Figueiredo, *As tecnologias da informação como instrumentos auxiliares do ensino secundário*, Folha Informativa do Projecto “Computação no Ensino da Matemática” **3** (1987).
- [14] L. Descalço e M. Aires, *A computer aided learn by example system*, International Conference in Mathematics Sciences and Science Education (2006).
- [15] M. Paula Oliveira e Sabrina Vieira da Silva, *An overview of pmate: developing software for all degrees of teaching*, International Conference in Mathematics Sciences and Science Education (2006).
- [16] Jaime Carvalho e Silva, *A matemática, a tecnologia e a escola*, Educação e Matemática **71** (2003), 1–2.
- [17] Carlos Nogueira Fino, *Uma turma da “geração nintendo” construindo uma cultura escolar nova*, Actas da II Conferência Internacional de Tecnologias da Informação e Comunicação na Educação (2001).
- [18] Enrique Hernández-Manfredini, *Linguagens regulares e autómatos*, Departamento de Matemática da Universidade de Aveiro, 1999.
- [19] J. E. Hopcroft and J. D. Ullman, *Introduction to automata theory, languages, and computation*, Addison-Wesley, 1979.
- [20] J. M. Howie, *Automata and languages*, Oxford University Press, 1991.
- [21] C. Fino J. Sousa, *As tic abrindo caminho a um novo paradigma educacional*, Actas do VI Congresso galaico-português de Psicopedagogia (2001), 371–381.

- [22] Sowmya Ramachandran James Ong, *Intelligent tutoring systems: using ai to improve training performance and roi*, Networker Newsletter **19** (2003).
- [23] Sowmya Ramachandran Jim Ong, *Intelligent tutoring systems: the what and the how*, Learning Circuits on-line magazine (2000).
- [24] Andrew McCallum e Fernando Pereira John Lafferty, *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*, Proceedings of the 8th International Conference on Machine Learning 2001.
- [25] Erik Haugsjaa Joseph Beck, Mia Stern, *Applications of ai in education*, Crosswords, the ACM Student Magazine.
- [26] M. Sidrach-de-Cardona e F. Triguero L. Mora-López, R. Morales-Bueno, *Probabilistic finite automata and randomness in nature: a new approach in the modelling and prediction of climate parameters*, International Environmental Modelling and Software Congress (2002).
- [27] Teresa B. Ludermir Marcilio C. P. de Souto and Wilson R. de Oliveira, *Equivalence between ram-based neural networks and probabilistic automata*, IEEE transactions on neural networks **16** (2005).
- [28] Émile Durkheim, *Educação e sociologia*, Edições 70, 2001.
- [29] Azaria Paz, *Introduction to probabilistic automata*, Academic Press, 1971.
- [30] F. Denis Pierre Dupont and Y. Esposito, *Links between probabilistic automata and hidden markov models: probability distributions, learning models and induction algorithms*, Special Issue on Gramatical Inference Techniques and Applications **38** (2004), 1349–1371.
- [31] Tania Mezzadri Centeno e Gilles Celleron Rodrigo Mikosz Goncalves, *Modelagem preditiva a partir de uma sequência de imagens de satélite utilizando autómatos celulares*, XI Simpósio Brasileiro de Sensoriamento Remoto **1** (2003), 975–981.
- [32] João Paulo Schwarz Schuler, *Simulação discreta, autómatos celulares e origem da vida*, (2000).

- [33] Branca Silveira, *Conversando com João Correia Freitas [11/11/2005]*, Educação e Matemática **86** (2005), 44–48.
- [34] A. Toffler, *A terceira vaga*, Lisboa: Edições Livros do Brasil, 1984.
- [35] Salih Usun, *Applications and problems of computer assisted education in turkey*, Turkish online journal of Educational Technology **5** (2006), 11–15.

Código C#

83


```

        "Integrais de superfície"
    };

// TemaObj -> Objectivos em cada tema (posição 0 = numero de objectivos no tema)
public int[,] TemaObj = {
    {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {9, 1, 2, 3, 4, 5, 9, 12, 13, 14, 0},
    {8, 6, 7, 8, 10, 11, 15, 16, 17, 0, 0},
    {5, 18, 19, 20, 21, 40, 0, 0, 0, 0, 0},
    {6, 22, 23, 24, 25, 26, 41, 0, 0, 0, 0},
    {7, 27, 28, 29, 30, 31, 38, 39, 0, 0, 0},
    {6, 32, 33, 34, 35, 36, 37, 0, 0, 0, 0}
};

// ObjTema -> criado a partir de TemaObj, tema de cada objectivo
public int[] ObjTema;
// Designação dos objectivos secundários
public string[] Indentif = { "",
    "774 Domínios e gráficos de f.v.v. 2206 Domínios e gráficos de f.v.v.",
    "775 limites 2212 limites",
    "776 Continuidade 2213 Continuidade num ponto",
    "776 Continuidade 2216 Teorema de Weierstrass",
    "777 Limites e continuidade de f.v.v. reais 2218 Limites e continuidade de f.v.v. reais",
    "781 Função de classe  $C^k$  2222 Função de classe  $C^k$ ",
    "784 O vector gradiente 2225 O vector gradiente",
    "787 Derivadas direccionais 2228 Derivadas direccionais",
    "789 Continuidade (f. vectoriais) 2233 Continuidade num ponto",
    "794 Derivada da função composta 2242 Derivada da função composta",
    "795 Diferenciabilidade 2243 Diferenciabilidade",
    "797 Noção de distância em  $R^n$  2245 Noção de distância em  $R^n$ ",
    "809 Conjuntos abertos e fechados 2272 Conjuntos abertos e fechados",
    "812 Conjuntos limitados e compactos 2276 Conjuntos limitados e compactos",
    "821 Fórmula de Taylor e extremos locais para uma função de 2 variáveis  
2295 Fórmula de Taylor de ordem 2 para uma função dada",
    "823 Determinar máximo e mínimos locais de uma função real de várias  
variáveis sujeita a uma restrição em igualdade 2303 Determinar máximo  
e mínimos locais de uma função real de várias variáveis sujeita  
a uma restrição em igualdade",
    "818 Função implícita real de variável real 2345 Função implícita  
real de diversas variáveis",
    "830 Integrais de linha no plano 2352 Definição",
    "830 Integrais de linha no plano 2353 Propriedades",
    "830 Integrais de linha no plano 2354 Mudança do integral duplo para o integral simples",
    "830 Integrais de linha no plano 2355 Mudança de variável",
    "828 Integrais triplos 2357 Definição",
    "828 Integrais triplos 2358 Propriedades 1186",
    "828 Integrais triplos 2359 Mudança do integral triplo para o integral simples",
    "828 Integrais triplos 2360 Mudança de variável",
    "828 Integrais triplos 2361 Aplicações",
    "829 Campos no plano e no espaço 2363 Campos no espaço",
    "830 Integrais de linha no plano 2365 Integral de uma função real",
    "830 Integrais de linha no plano 2366 Aplicações do integral de funções reais",
};

```



```

"830 Integrais de linha no plano 2367 Integral de um campo no plano",
"830 Integrais de linha no plano 2370 Teorema de Green",
"831 Integrais de superfície 2371 Parametrizações de superfícies",
"831 Integrais de superfície 2372 Integral de superfície de funções reais",
"831 Integrais de superfície 2373 Fórmula de cálculo do integral de superfície",
"831 Integrais de superfície 2375 Integral de superfície de campos no espaço",
"831 Integrais de superfície 2376 Aplicações do integral de superfície de campos
    no espaço",
"831 Integrais de superfície 2377 Teorema da divergência ou de Gauss",
"832 Integrais de linha no espaço e Teorema de Stokes 2378 Parametrizações de linhas",
"832 Integrais de linha no espaço e Teorema de Stokes 2384 Teorema de Stokes",
"827 Integrais duplos 2399 Integrais duplos",
"828 Integrais triplos 2400 Integrais triplos"
};

// Modelos em cada objectivo secundário
public int[,] ModOS = {
    {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {3, 916, 934, 935, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 979, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 946, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 961, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 889, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 940, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 884, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 911, 915, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 941, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {3, 914, 939, 944, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 883, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 888, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 885, 887, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 886, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 943, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 912, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 913, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 1194, 1195, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1167, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 1165, 1166, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1168, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1220, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1186, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1187, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1221, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1213, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 1008, 1013, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 1005, 1006, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1040, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 1041, 1118, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {3, 1042, 1146, 1147, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 983, 1072, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {2, 991, 992, 0, 0, 0, 0, 0, 0, 0, 0, 0},
    {1, 1011, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},

```

```

        {4, 993,    994,    1010,    1014,    0, 0, 0, 0, 0, 0},
        {1, 1071,    0, 0, 0, 0, 0, 0, 0, 0, 0},
        {2, 1012,    1073,    0, 0, 0, 0, 0, 0, 0, 0},
        {1, 995,    0, 0, 0, 0, 0, 0, 0, 0, 0},
        {2, 982,    1009,    0, 0, 0, 0, 0, 0, 0, 0},
        {1, 1119,    0, 0, 0, 0, 0, 0, 0, 0, 0},
        {1, 1193,    0, 0, 0, 0, 0, 0, 0, 0, 0}
    };

    // "conhecimento" = conhecimento a atingir em cada objectivo
    public int conhecimento = 90;
    // número de objectivos não atingidos tolerados para paragem
    public int tolerancia = 35;
    public ClasseDisciplina()
    {
        int pmin = pn; pmed=pd; pmax=px;
        // Inicia ObjTema
        ObjTema = new int[SizeMatr];
        int i,l,t,j;
        for (l=1; l<=numobj;l++)
        {
            t=0;
            i=1;
            while (t==0)
            {
                j=1;
                while(t==0 && j<=TemaObj[i,0])
                {
                    if (l == TemaObj[i,j] ) t=i;
                    j++;
                }
                ObjTema[l] = t;
                i++;
            }
        }
        //
        // TODO: Add constructor logic here
        //
    }
    public double[] IniciaP()
    {
        return P_0;
    }
    public string Strvardisc()
    {
        return vardisc.ToString();
    }
}

// ClasseAluno vai ser dividida em duas: ClasseAluno e ClasseDisciplina
// É preciso substituir as chamadas a "Aluno.xxx()" por "Aluno.xxx(Dis)" na webform
// aqui dentro da classe aluno basta escrever Dis.A em vez de A quando A passar para Dis

```

```

// Para inserir modelo é preciso alterar apenas "Mod0s"
// Para inserir objectivo é preciso alterar: A, B, I, Mod0s, Identif, TemaObj
using System;
namespace sa2cv001 {
    /// <summary>
    /// Summary description for ClasseAluno.
    /// </summary>
    public class ClasseAluno
    {
        ClasseDisciplina Dis;
        const int SizeMatr = 51;
        // Declaração das variáveis globais
        double[] P;
        int[] C;
        int[] V;
        // OBJ-> índice do objectivo secundário actual
        int OBJ;
        // IdModelo -> Id do proximo modelo
        int IdModelo;
        // pesos relativos entre "C-> conhecimento", "V-> permanencia", "A-> proximidade", "B-> bases"
        double pesoC = 10.0, pesoV = 0.0, pesoA = 90.0, pesoB = 0.0;
        // incrementos ao conhecimento conforme o número de respostas correctas
        int resp4 = 20, resp3 = 10, resp2 = 0, resp1 = -10, resp0 = -20;
        // variável com o valor lógico da condição de paragem
        int terminar = 0; // false
        // número total de passagens no ciclo
        int contador = 0;
        // número de objectivos atingidos
        int objating;
        // Objecto para os numeros aleatorios
        Random Aleat;
        // paramatros para a funcao PARAMETROS, a qual actualiza os parametros para
        // o cálculo das probabilidades
        int tema_perc_min=90; // conhecimento minimo em cada objectivo do tema
        int perm_max = 3; // maximo de permanencia no tema = perm_max * (num obj do tema)
        double tema_con_min = 0.6; // percentagem de objectivos exigida no tema
        double con_alto = 0.6; // percentagem de objectivos com conhecimento minimo
        // para considerar o conhecimento na disciplina elevado
        // decidindo entao aumentar pesoC e nao pesoV
        public ClasseAluno(ClasseDisciplina Dis)
        {
            //
            // TODO: Add constructor logic here
            //
            // Variáveis auxiliares
            int aux;
            int i;
            double N;
            int semente;
            double k;
            int classif;

```

```

        Aleat = new Random();
        C = new int[SizeMatr];
        // iniciar C-> conhecimento
        for (i= 1; i <= Dis.numobj; ++i)
            C[i] = 50;
        // iniciar V-> permanencia
        V = new int[SizeMatr];
        for (i=1; i <= Dis.numobj; ++i)
            V[i]= 0;
        // iniciar P-> P(i) = probabilidade de ser seleccionado o objectivo secundário i
        P = new double[SizeMatr];
        P = Dis.IniciaP();
        // Escolhe, aleatoriamente com base em P, um objectivo (número entre 1 e numobj)
        N = (double) Aleat.Next(0,10000) / 10000;
        OBJ =1;
        k = P[OBJ];
        while (N > k)
        {
            OBJ=OBJ+1;
            k = k + P[OBJ];
        }
        IdModelo = (int) Dis.ModOS[OBJ,Aleat.Next(1,Dis.ModOS[OBJ,0])];
    } // fim de ClasseAluno()
    // Funcao Resp - é chamada após cada resposta do aluno - actualiza tudo
    public void Resp(ClasseDisciplina Dis, int classif)
    {
        int aux;
        // actualização do vector "permanência"
        V[OBJ] = V[OBJ]+1;
        // actualização do vector "conhecimento"
        if (classif == 4)
            C[OBJ] = min(C[OBJ]+resp4,100);
        else if (classif == 3)
            C[OBJ] = min(C[OBJ]+resp3,100);
        else if (classif == 2)
            C[OBJ] = C[OBJ]+resp2;
        else if (classif == 1)
            C[OBJ] = max(C[OBJ]+resp1,0);
        else if (classif == 0)
            C[OBJ] = max(C[OBJ]+resp0,0);
        // actualizar parametros
        aux = PARAMETROS(Dis);
        // actualizar probabilidades
        aux = PROBABILIDADES(Dis);
        // transição para novo objectivo
        OBJ = TRANSICAO(Dis);
        // escolhe novo modelo com objectivo OBJ
        IdModelo = SetIdModelo(Dis);
    }
    private int min(int n, int m)
    {

```

```

        if (n > m)
            return m;
        else
            return n;
    }
private int max(int n, int m)
{
    if (n > m)
        return n;
    else
        return m;
}
public string StrConhec(ClasseDisciplina Dis)
{
    string aux;
    int i;
    aux = "";
    for (i=1; i<=Dis.numobj; i++)
    {
        aux = aux + C[i].ToString() + " ";
    }
    return aux;
}
public string StrConhecTema(ClasseDisciplina Dis)
{
    string aux;
    int i;
    aux = "";
    for (i=1; i<=Dis.numobj; i++)
    {
        if (Dis.ObjTema[OBJ] == Dis.ObjTema[i])
            aux = aux + C[i].ToString() + " ";
    }
    return aux;
}
public string StrOBJ(ClasseDisciplina Dis)
{
    return Dis.I[OBJ].ToString();
}
public string StrIdModelo(ClasseDisciplina Dis)
{
    return IdModelo.ToString();
}
public int RetIdModelo(ClasseDisciplina Dis)
{
    return IdModelo;
}
public int SetIdModelo(ClasseDisciplina Dis)
{
    return (int) Dis.ModOS[OBJ,Aleat.Next(1,Dis.ModOS[OBJ,0])];
}

```

```

public string StrObjating(ClasseDisciplina Dis)
{
    return objating.ToString();
}
public string Objectivo(ClasseDisciplina Dis)
{
    return Dis.Indentif[OBJ];
}
public string Tema(ClasseDisciplina Dis)
{
    return Dis.Temas[Dis.ObjTema[OBJ]];
}
public string StrDisvardisc(ClasseDisciplina Dis)
{
    return Dis.vardisc.ToString();
}
public int Terminar(ClasseDisciplina Dis)
{
    // actualiza variável de paragem "terminar"
    int i;

    for (i=1; i <= Dis.numobj; i++)
        if (C[i] >= Dis.conhecimento)
            objating = objating + 1;
    if (objating > Dis.numobj - Dis.tolerancia)
        // Environment.Exit(...);
        return 1;
    else
        return 0;
}
// -----
// FUNÇÃO PARAMETROS
//
// Actualiza os parametros usados no calculo das probabilidades:
public int PARAMETROS(ClasseDisciplina Dis)
{
    int tema = Dis.ObjTema[OBJ];
    int csta; // numero de objectivo com conhecimento satisfario no tema actual
    int cs; // numero de objectivo com conhecimento satisfario na disciplina
    int pta; // permanencia total no tema actual
    int i;
    csta = 0; pta = 0; cs = 0;
    for(i=1; i<=Dis.TemaObj[tema,0]; i++)
    {
        pta = pta + V[Dis.TemaObj[tema,i]];
        if(C[Dis.TemaObj[tema,i]]>=tema_perc_min) csta=csta+1;
    }
    for(i=1; i<=Dis.numobj; i++)
    {
        if (C[i] >= tema_perc_min) cs = cs+1;
    }
}

```

```

if (csta < tema_con_min*Dis.TemaObj[tema,0])
{
    if (pta>perm_max*Dis.TemaObj[tema,0])
    {
        pesoC=10.0; pesoV=20.0; pesoA=20.0; pesoB=50.0;
    }
    else
    {
        pesoC=10.0; pesoV=0.0; pesoA=70.0; pesoB=20.0;
    }
}
else
if (cs >= con_alto * Dis.numobj)
{
    pesoC=90.0; pesoV=10.0; pesoA=0.0; pesoB=0.0;
}
else
{
    pesoC=70.0; pesoV=0.0; pesoA=30.0; pesoB=0.0;
}
return(1);
}
// -----
// FUNÇÃO PROBABILIDADES
// %
// Atualiza o vector P com base em A, B, C e V
public int PROBABILIDADES(ClasseDisciplina Dis)
{
    // Atualiza probabilidades: vetor P.
    // Usa variáveis globais: P, A, B, C, V, numobj;
    int[] Anum;
    int[] Bnum;
    int[] Cnum;
    int[] Vnum;
    int SomaA, SomaB, SomaC, SomaV, totA, totB, totC,totV;
    int i,maxv;
    // SomaA -> contém a soma da linha da matriz A correspondente a OBJ
    SomaA = 0;
    for (i=1; i <= Dis.numobj; ++i)
        SomaA = SomaA + Dis.A[OBJ,i];
    // SomaB -> contém a soma da linha da matriz B correspondente a OBJ
    SomaB =0;
    for (i=1; i <= Dis.numobj; ++i)
        SomaB = SomaB + Dis.B[OBJ,i];

    // SomaC -> contém a soma do vector C
    SomaC =0;
    for (i=1; i <= Dis.numobj; ++i)
        SomaC =SomaC + C[i];

    // SomaV -> contém a soma do vector V

```

```

SomaV=0;
for (i=1; i <= Dis.numobj; ++i)
    SomaV = SomaV + V[i];

Anum = new int[SizeMatr];

// contribuição de A: Anum / totA
for (i=1; i <= Dis.numobj; ++i)
    Anum[i] = Dis.A[OBJ,i];
totA = SomaA;

Bnum = new int[SizeMatr];
// contribuição de B: Bnum / totB
if (C[OBJ] < 50)
{
    for (i=1; i <= Dis.numobj; ++i)
        Bnum[i] = Dis.B[OBJ,i];
    totB = SomaB;
}
else
{
    for (i=1; i <= Dis.numobj; ++i)
        Bnum[i] = 1-Dis.B[OBJ,i];
    totB = Dis.numobj-SomaB;
}
Cnum = new int[SizeMatr];
// contribuição de C: Cnum / totC
for (i=1; i <= Dis.numobj; ++i)
    Cnum[i] = 100-C[i];
totC = 100*Dis.numobj-SomaC;
Vnum = new int[SizeMatr];
// contribuição de V: Vnum / totV
maxv =0;
for (i=1; i <= Dis.numobj; ++i)
    if (V[i] > maxv)
        maxv = V[i];
for (i=1; i <= Dis.numobj; ++i)
    Vnum[i] = maxv - V[i];
totV = maxv*Dis.numobj-SomaV;
// PPP Atualização das probabilidades
for (i=1; i <= Dis.numobj; ++i)
    P[i]=(pesoA*Anum[i] + pesoB*Bnum[i] + pesoC*Cnum[i] + pesoV*Vnum[i])/
        (pesoA*totA + pesoB*totB + pesoC*totC + pesoV*totV);
return(1);
}
// Fim: FUNÇÃO PROBABILIDADES
// -----
// FUNÇÃO TRANSIÇÃO
//
// Escolhe, aleatoriamente com base em P, um objectivo (número entre 1 e numobj)
public int TRANSICAO(ClasseDisciplina Dis)

```



```

    {
        // Usa variável global P, devolve novo indice de objectivo
        double N;
        double k;
        int I =1;
        // N = rand()*1.0/RAND_MAX; /* número aleatório entre 0 e 1 */
        N = (double)Aleat.Next(0,10000) / 10000;
        k = P[I];
        while (N > k)
        {
            I=I+1;
            k = k + P[I];
        }
        return I;
    }
    // Fim: FUNÇÃO TRANSIÇÃO
}

}

\newpage
{\bf \% File "ClasseAluno.cs"}
// ClasseAluno vai ser dividida em duas: ClasseAluno e ClasseDisciplina
// É preciso substituir as chamadas a "Aluno.xxx()" por "Aluno.xxx(Dis)" na webform
// aqui dentro da classe aluno basta escrever Dis.A em vez de A quando A passar para Dis
// Para inserir modelo é preciso alterar apenas "Mod0s"
// Para inserir objectivo é preciso alterar: A, B, I, Mod0s, Identif, TemaObj
using System;
namespace sa2cv001 {
    /// <summary>
    /// Summary description for ClasseAluno.
    /// </summary>
    public class ClasseAluno
    {
        ClasseDisciplina Dis;
        const int SizeMatr = 51;
        // Declaração das variáveis globais
        double[] P;
        int[] C;
        int[] V;
        // OBJ-> índice do objectivo secundário actual
        int OBJ;
        // IdModelo -> Id do proximo modelo
        int IdModelo;
        // pesos relativos entre "C-> conhecimento", "V-> permanencia", "A-> proximidade", "B-> bases"
        double pesoC = 10.0, pesoV = 0.0, pesoA = 90.0, pesoB = 0.0;
        // incrementos ao conhecimento conforme o número de respostas correctas
        int resp4 = 20, resp3 = 10, resp2 = 0, resp1 = -10, resp0 = -20;
        // variável com o valor lógico da condição de paragem
        int terminar = 0; // false
        // número total de passagens no ciclo
        int contador = 0;
        // número de objectivos atingidos
    }
}

```

```

int objating;
// Objecto para os numeros aleatorios
Random Aleat;
// paramatros para a funcao PARAMETROS, a qual actualiza os parametros para
// o cálculo das probabilidades
int tema_perc_min=90; // conhecimento minimo em cada objectivo do tema
int perm_max = 3; // maximo de permanencia no tema = perm_max * (num obj do tema)
double tema_con_min = 0.6; // percentagem de objectivos exigida no tema
double con_alto = 0.6; // percentagem de objectivos com conhecimento minimo
                        // para considerar o conhecimento na disciplina elevado
                        // decidindo entao aumentar pesoC e nao pesoV
public ClasseAluno(ClasseDisciplina Dis)
{
    //
    // TODO: Add constructor logic here
    //
    // Variáveis auxiliares
    int aux;
    int i;
    double N;
    int semente;
    double k;
    int classif;
    Aleat = new Random();
    C = new int[SizeMatr];
    // iniciar C-> conhecimento
    for (i= 1; i <= Dis.numobj; ++i)
        C[i] = 50;
    // iniciar V-> permanencia
    V = new int[SizeMatr];
    for (i=1; i <= Dis.numobj; ++i)
        V[i]= 0;
    // iniciar P-> P(i) = probabilidade de ser seleccionado o objectivo secundário i
    P = new double[SizeMatr];
    P = Dis.IniciaP();
    // Escolhe, aleatoriamente com base em P, um objectivo (número entre 1 e numobj)
    N = (double) Aleat.Next(0,10000) / 10000;
    OBJ =1;
    k = P[OBJ];
    while (N > k)
    {
        OBJ=OBJ+1;
        k = k + P[OBJ];
    }
    IdModelo = (int) Dis.ModOS[OBJ,Aleat.Next(1,Dis.ModOS[OBJ,0])];
} // fim de ClasseAluno()
// Funcao Resp - é chamada após cada resposta do aluno - actualiza tudo
public void Resp(ClasseDisciplina Dis, int classif)
{
    int aux;
    // actualização do vector "permanência"

```

```

V[OBJ] = V[OBJ]+1;
// actualização do vector "conhecimento"
if (classif == 4)
    C[OBJ] = min(C[OBJ]+resp4,100);
else if (classif == 3)
    C[OBJ] = min(C[OBJ]+resp3,100);
else if (classif == 2)
    C[OBJ] = C[OBJ]+resp2;
else if (classif == 1)
    C[OBJ] = max(C[OBJ]+resp1,0);
else if (classif == 0)
    C[OBJ] = max(C[OBJ]+resp0,0);
// actualizar parametros
aux = PARAMETROS(Dis);
// actualizar probabilidades
aux = PROBABILIDADES(Dis);
// transição para novo objectivo
OBJ = TRANSICAO(Dis);
// escolhe novo modelo com objectivo OBJ
IdModelo = SetIdModelo(Dis);
}
private int min(int n, int m)
{
    if (n > m)
        return m;
    else
        return n;
}
private int max(int n, int m)
{
    if (n > m)
        return n;
    else
        return m;
}
public string StrConhec(ClasseDisciplina Dis)
{
    string aux;
    int i;
    aux = "";
    for (i=1; i<=Dis.numobj; i++)
    {
        aux = aux + C[i].ToString() + " ";
    }
    return aux;
}
public string StrConhecTema(ClasseDisciplina Dis)
{
    string aux;
    int i;
    aux = "";

```

```

        for (i=1; i<=Dis.numobj; i++)
        {
            if (Dis.ObjTema[OBJ] == Dis.ObjTema[i])
                aux = aux + C[i].ToString() + " ";
        }
        return aux;
    }
    public string StrOBJ(ClasseDisciplina Dis)
    {
        return Dis.I[OBJ].ToString();
    }
    public string StrIdModelo(ClasseDisciplina Dis)
    {
        return IdModelo.ToString();
    }
    public int RetIdModelo(ClasseDisciplina Dis)
    {
        return IdModelo;
    }
    public int SetIdModelo(ClasseDisciplina Dis)
    {
        return (int) Dis.ModOS[OBJ,Aleat.Next(1,Dis.ModOS[OBJ,0])];
    }
    public string StrObjating(ClasseDisciplina Dis)
    {
        return objating.ToString();
    }
    public string Objectivo(ClasseDisciplina Dis)
    {
        return Dis.Indentif[OBJ];
    }
    public string Tema(ClasseDisciplina Dis)
    {
        return Dis.Temas[Dis.ObjTema[OBJ]];
    }
    public string StrDisvardisc(ClasseDisciplina Dis)
    {
        return Dis.vardisc.ToString();
    }
    public int Terminar(ClasseDisciplina Dis)
    {
        // atualiza variável de paragem "terminar"
        int i;
        for (i=1; i <= Dis.numobj; i++)
            if (C[i] >= Dis.conhecimento)
                objating = objating + 1;
        if (objating > Dis.numobj - Dis.tolerancia)
            // Environment.Exit(...);
            return 1;
        else
            return 0;
    }

```

```

}
// -----
// FUNÇÃO PARAMETROS
//
// Actualiza os parametros usados no calculo das probabilidades:

public int PARAMETROS(ClasseDisciplina Dis)
{
    int tema = Dis.ObjTema[OBJ];
    int csta; // numero de objectivo com conhecimento satisfario no tema actual
    int cs; // numero de objectivo com conhecimento satisfario na disciplina
    int pta; // permanencia total no tema actual
    int i;
    csta = 0; pta = 0; cs = 0;
    for(i=1; i<=Dis.TemaObj[tema,0]; i++)
    {
        pta = pta + V[Dis.TemaObj[tema,i]];
        if(C[Dis.TemaObj[tema,i]]>=tema_perc_min) csta=csta+1;
    }
    for(i=1; i<=Dis.numobj; i++)
    {
        if (C[i] >= tema_perc_min) cs = cs+1;
    }
    if (csta < tema_con_min*Dis.TemaObj[tema,0])
    {
        if (pta>perm_max*Dis.TemaObj[tema,0])
        {
            pesoC=10.0; pesoV=20.0; pesoA=20.0; pesoB=50.0;
        }
        else
        {
            pesoC=10.0; pesoV=0.0; pesoA=70.0; pesoB=20.0;
        }
    }
    else
    if (cs >= con_alto * Dis.numobj)
    {
        pesoC=90.0; pesoV=10.0; pesoA=0.0; pesoB=0.0;
    }
    else
    {
        pesoC=70.0; pesoV=0.0; pesoA=30.0; pesoB=0.0;
    }
    return(1);
}
// -----
// FUNÇÃO PROBABILIDADES
// %
// Actualiza o vector P com base em A, B, C e V
public int PROBABILIDADES(ClasseDisciplina Dis)
{

```

```

// Atualiza probabilidades: vetor P.
// Usa variáveis globais: P, A, B, C, V, numobj;
int[] Anum;
int[] Bnum;
int[] Cnum;
int[] Vnum;
int SomaA, SomaB, SomaC, SomaV, totA, totB, totC, totV;
int i, maxv;
// SomaA -> contém a soma da linha da matriz A correspondente a OBJ
SomaA = 0;
for (i=1; i <= Dis.numobj; ++i)
    SomaA = SomaA + Dis.A[OBJ,i];
// SomaB -> contém a soma da linha da matriz B correspondente a OBJ
SomaB = 0;
for (i=1; i <= Dis.numobj; ++i)
    SomaB = SomaB + Dis.B[OBJ,i];
// SomaC -> contém a soma do vector C
SomaC = 0;
for (i=1; i <= Dis.numobj; ++i)
    SomaC = SomaC + C[i];
// SomaV -> contém a soma do vector V
SomaV = 0;
for (i=1; i <= Dis.numobj; ++i)
    SomaV = SomaV + V[i];
Anum = new int[SizeMatr];
// contribuição de A: Anum / totA
for (i=1; i <= Dis.numobj; ++i)
    Anum[i] = Dis.A[OBJ,i];
totA = SomaA;
Bnum = new int[SizeMatr];
// contribuição de B: Bnum / totB
if (C[OBJ] < 50)
{
    for (i=1; i <= Dis.numobj; ++i)
        Bnum[i] = Dis.B[OBJ,i];
    totB = SomaB;
}
else
{
    for (i=1; i <= Dis.numobj; ++i)
        Bnum[i] = 1-Dis.B[OBJ,i];
    totB = Dis.numobj-SomaB;
}
Cnum = new int[SizeMatr];
// contribuição de C: Cnum / totC
for (i=1; i <= Dis.numobj; ++i)
    Cnum[i] = 100-C[i];
totC = 100*Dis.numobj-SomaC;
Vnum = new int[SizeMatr];
// contribuição de V: Vnum / totV
maxv = 0;

```

```

        for (i=1; i <= Dis.numobj; ++i)
            if (V[i] > maxv)
                maxv = V[i];
        for (i=1; i <= Dis.numobj; ++i)
            Vnum[i] = maxv - V[i];
        totV = maxv*Dis.numobj-SomaV;

        // PPP Actualização das probabilidades
        for (i=1; i <= Dis.numobj; ++i)
            P[i]=(pesoA*Anum[i] + pesoB*Bnum[i] + pesoC*Cnum[i] + pesoV*Vnum[i])/
                (pesoA*totA + pesoB*totB + pesoC*totC + pesoV*totV);
        return(1);
    }
    // Fim: FUNÇÃO PROBABILIDADES
    // -----
    // FUNÇÃO TRANSIÇÃO
    //
    // Escolhe, aleatoriamente com base em P, um objectivo (número entre 1 e numobj)
    public int TRANSICAO(ClasseDisciplina Dis)
    {
        // Usa variável global P, devolve novo indice de objectivo
        double N;
        double k;
        int I =1;
        // N = rand()*1.0/RAND_MAX; /* número aleatório entre 0 e 1 */
        N = (double)Aleat.Next(0,10000) / 10000;
        k = P[I];
        while (N > k)
        {
            I=I+1;
            k = k + P[I];
        }
        return I;
    }
    // Fim: FUNÇÃO TRANSIÇÃO
}
}

```

Apêndice B

Tabelas

Category: Utilization		
Sub-Category	Indicators	2001
Teachers	Percentage of subjects utilizing multimedia materials for teaching and learning through WWW (%)	50.08
	Percentage of classes using webboard (%)	39.48
	Percentage of subjects using webboard (%)	57.19
	Percentage of teachers participating in an association regarding ICT use (%)	13.40
	Percentage of subjects using computer lab (%)	9.29
	Percentage of functions of the Information System being used (%)	33.14
	Whether to use electronic administrative system (Yes=1, No=0)	0.16
	Number of posts written for and by parents	0.41
	Hours of use of computer per a student (hrs/1yr.)	6.36
	Percentage of students participating special programs after school regarding ICT (Elementary)(%)	38.34
Students	Whether to teach about computer (Secondary)(Yes=1, No=0)(%)	23.00
	Percentage of subjects that teachers collect assignments through webboard (%)	21.53
	Number of students searching library DB per day	113.64
	Percentage of student-governing community utilizing web environment (%)	95.28
	Whether to have a program regarding Internet ethics (%)	0.76
	Number of web communities per class	2.65
Internet	Usage of Internet (Kb/Second)	1910.83
	Number of Access to school website per week	0.71
	How often to revise school website (days)	101.73
Category: Outputs		
Sub-Category	Indicators	2001
Teachers	Percentage of teachers having email address (%)	96.10
	Percentage of teachers having homepage (%)	18.45
	Percentage of teachers having certificates regarding ICT (%)	29.35
	Percentage of teachers participating educational S/W contests (%)	13.24
Students	Percentage of students having email address (%)	83.76
	Percentage of students having homepage (%)	9.59
	Percentage of students having certificates regarding ICT (%)	2.98
	Percentage of students participating educational S/W contests (%)	51.73
	Percentage of students completing the ICT course(32hours) (%)	11.82

Figura B.1: Tabela com dados relativos ao uso das novas tecnologias nas salas de aula da Coreia.



Information and Communications Technologies (ICTs) in Schools

Percentage of teachers who have used computers in class in the last 12 months in Europe

	Total	Primary schools	Lower secondary schools	Upper secondary schools	Vocational schools
BELGIUM	69.0	66.9	73.7	74.8	78.3
CZECH REPUBLIC	78.3	82.4	78.9	69.5	71.0
DENMARK	94.6	95.7	94.4	97.8	93.5
GERMANY	78.0	78.0	77.2	80.4	78.6
ESTONIA	59.7	60.9	61.5	53.3	46.8
GREECE	35.6	32.8	38.0	44.1	58.0
SPAIN	68.2	68.9	66.6	65.5	67.5
FRANCE	65.5	65.7	56.1	72.1	78.9
IRELAND	81.7	86.5	64.1	64.2	69.9
ITALY	72.4	71.6	71.9	72.4	81.6
CYPRUS	75.0	87.2	39.8	50.7	58.2
LATVIA	34.9	35.7	37.8	33.6	27.7
LITHUANIA	59.3	58.7	65.0	64.1	74.3
LUXEMBOURG	70.2	74.4	54.2	43.6	61.8
HUNGARY	42.8	36.8	40.1	60.0	64.1
MALTA	74.5	82.6	59.1	59.1	76.7
THE NETHERLANDS	90.0	91.7	80.9	77.4	84.0
AUSTRIA	87.9	87.9	87.5	81.3	86.1
POLAND	61.4	60.2	60.3	67.1	70.9
PORTUGAL	69.5	70.0	66.2	71.7	75.2
SLOVENIA	67.6	71.7	71.6	53.6	52.2
SLOVAKIA	70.3	72.0	73.1	65.5	69.7
FINLAND	85.1	88.0	77.1	80.5	81.4
SWEDEN	90.9	90.0	91.7	94.6	87.7
UK	96.4	97.4	90.4	91.5	92.9
ICELAND	79.5	78.6	84.7	84.2	83.3
NORWAY	89.4	90.4	89.4	79.4	82.4
EU25	74.3	75.2	70.9	73.0	76.7
EU25+2	74.5	75.3	71.2	73.0	76.7
EU15	77.2	78.0	74.8	76.1	78.8
10 NEW MEMBER STATES	61.3	60.6	59.5	64.1	68.1
Base:	All teachers				
Question:	q7				
Wording:	How have you used computers and/or the internet for work in the last 12 months?... you used a computer and/or the internet in class* while teaching				
Indicator:	Percentage of teachers who used computers in class in the last 12 months				
Source:	empirica: LearnInd 2006 (CTS)				

Figura B.2: Tabela com a percentagem de professores que usou o computador na sala de aula no ano de 2006 nos diferentes tipos de escolas dos diversos países da União Europeia.

Total number of internet computers per 100 pupils by school type

	Total	Primary schools	Lower secondary schools	Upper secondary schools	Vocational schools
BELGIUM	7.7	5.2	12.0	10.6	12.0
CZECH REPUBLIC	8.2	6.4	6.2	10.1	11.0
DENMARK	26.3	17.9	17.9	37.2	48.0
GERMANY	7.7	7.1	7.4	7.5	8.7
ESTONIA	7.2	6.1	5.9	6.4	13.8
GREECE	5.9	4.2	6.1	8.3	19.1
SPAIN	8.5	7.4	9.2	10.6	11.2
FRANCE	8.9	4.8	10.0	14.0	18.2
IRELAND	8.7	7.0	8.9	9.9	14.0
ITALY	6.5	4.2	5.1	9.8	10.5
CYPRUS	8.9	6.1	9.9	12.1	9.6
LATVIA	5.1	4.8	4.9	4.8	5.8
LITHUANIA	5.2	4.7	5.2	5.4	7.8
LUXEMBOURG	18.3	20.9	21.3	18.9	7.9
HUNGARY	8.6	5.9	7.1	11.0	14.8
MALTA	10.2	12.2	8.6	8.6	10.2
THE NETHERLANDS	20.0	14.2	18.4	20.4	26.8
AUSTRIA	14.2	7.0	12.8	19.8	23.1
POLAND	5.6	4.7	5.2	7.0	6.8
PORTUGAL	5.4	4.4	5.2	6.2	15.3
SLOVENIA	7.5	7.4	7.5	7.8	8.5
SLOVAKIA	5.8	4.9	4.5	6.9	7.4
FINLAND	16.2	11.3	11.9	17.2	21.7
SWEDEN	16.5	13.8	12.7	27.5	17.0
UK	18.5	14.6	23.4	24.9	28.4
ICELAND	14.8	14.1	14.3	17.2	17.6
NORWAY	22.7	16.0	20.5	39.9	38.6
EU25	9.9	7.7	9.7	11.4	14.1
EU25+2	10.0	7.8	9.8	11.6	14.3
EU15	10.6	8.2	10.7	12.4	15.1
10 NEW MEMBER STATES	6.4	5.3	5.8	7.7	8.9
USA*(2003)	4.4	4.9	3.8	3.8	---

Base: All schools of the respective breakdown category and country.
Questions Q4; Q6
Wording: Q4: How many pupils does your school have ?
Q6: In your school, how many computers are used for educational purposes for pupils, either to use alone or with a teacher? Please do not include computers that are only accessible to teachers or staff members.
Q11: Of the computers used for educational purposes by pupils, either alone or with a teacher, approximately how many are connected to the internet?
Indicator: Total number of internet computers (in breakdown aggregate) / total number of pupils (in breakdown aggregate) *100. (Note: not a school average, but aggregate estimator)
Source: empirica: LearnInd 2006 (HTS)
* USA data source: Internet Access in U.S. Public Schools and Classrooms: 1994-2003, Institute for education science, US department of education, February 2005

Figura B.3: Tabela com o número de computadores ligados à Internet por cada 100 alunos nos diferentes tipos de escolas dos diversos países da União Europeia.

Apêndice C

Modelos

Matemática

CICLO	TEMA-ID	OS	ND	CNT
-------	---------	----	----	-----

Latex actualizado a

Latex imprimido a 14 de Fevereiro de 2007

Identificação do Modelo

Área	Matemática
Id do modelo	
Objectivo Secundário	2233
Informação adicional	Limites e continuidade de funções vectoriais.
Tipo de Modelo	0 - Texto + Respostas com MathML
Ciclo de Ensino	5
Nível de Dificuldade	2

Objectivos das Respostas

R_1	Objmicro 5151	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_2	Objmicro 5151	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_3	Objmicro 5151	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	

R_4	Objmicro 5151	Peso 1
	Objmicro	Peso
	Objmicro	Peso
R_5	Objmicro 5151	Peso 1
	Objmicro	Peso
	Objmicro	Peso

Historial

Elaborado por	Luis Descalço e Maribel Aires	26 de Abril de 2005
Responsável	Luis Descalço	26 de Abri de 2005

Notas / Referências Bibliográficas

- Bortolossi, H.J., Cálculo Diferencial a Várias Variáveis, Uma Introdução à Teoria de Optimização, Ed. Loyola, 2002.
- Stewart, J., Cálculo, vol II, 4 ed., 2001.

MODELO

Observações/indicações de programação

--

Notações e Abreviaturas

$X = \begin{cases} (x, y) \text{ se } \bar{n} = 2 \\ (x, y, z) \text{ se } \bar{n} = 3 \end{cases}; \bar{p} = \begin{cases} (a, b) \text{ se } \bar{n} = 2 \\ (a, b, c) \text{ se } \bar{n} = 3 \end{cases}; \bar{f} = \begin{cases} f(x, y) \text{ se } \bar{n} = 2 \\ f(x, y, z) \text{ se } \bar{n} = 3 \end{cases};$
$\bar{E} = \begin{cases} f_1(x, y), f_2(x, y), \bar{n} = \bar{m} = 2 \\ f_1(x, y), f_2(x, y), f_3(x, y), \bar{n} = 2, \bar{m} = 3 \\ f_1(x, y, z), f_2(x, y, z), \bar{n} = 3, \bar{m} = 2 \\ f_1(x, y, z), f_2(x, y, z), f_3(x, y, z), \bar{n} = \bar{m} = 3 \end{cases}; \bar{U} = \begin{cases} (x_n), (y_n) \text{ se } \bar{n} = 2 \\ (x_n), (y_n), (z_n) \text{ se } \bar{n} = 3 \end{cases};$
$\bar{V} = \begin{cases} x_n, y_n \text{ se } \bar{n} = 2 \\ x_n, y_n, z_n \text{ se } \bar{n} = 3 \end{cases}; \bar{q} = \begin{cases} (r, s) \text{ se } \bar{m} = 2 \\ (r, s, t) \text{ se } \bar{m} = 3 \end{cases}; \bar{t} = \begin{cases} a, b \text{ se } \bar{n} = 2 \\ a, b, c \text{ se } \bar{n} = 3 \end{cases}; \bar{z}_1 = r; \bar{z}_2 = s; \bar{z}_3 = t.$

MODELO Domínio dos Parâmetros

$\bar{m}, \bar{n} \in \{2, 3\};$

Texto

Seja $f : D_f \subseteq \mathbb{R}^{\bar{n}} \rightarrow \mathbb{R}^{\bar{m}}$ uma função vectorial definida por $\bar{f} = (\bar{E})$.
--

Respostas

Verdadeira se e

só se:

R1	Dizemos que f é contínua no ponto $\bar{p} \in D_f$ se para quaisquer sucessões \bar{U} de pontos de D_f tivermos que $(\bar{V}) \rightarrow \bar{p}$ implica que $f(\bar{V}) \rightarrow f(\bar{p})$. c11		c11
	Dizemos que f é contínua no ponto $\bar{p} \in D_f$ se existem sucessões \bar{U} de pontos de D_f tais que $(\bar{V}) \rightarrow \bar{p}$ e $f(\bar{V}) \rightarrow f(\bar{p})$. c12		
R2	A função $f = \bar{f}$	é contínua em $\bar{p} \in D_f$ se e só se as suas funções coordenadas \bar{E} são contínuas em \bar{p} c11	c11
		pode ser contínua em $\bar{p} \in D_f$ mesmo que uma das suas funções coordenadas não seja contínua em \bar{p} c12	
		pode não ser contínua em $\bar{p} \in D_f$ mesmo que todas as suas funções coordenadas sejam contínuas em \bar{p} c13	

R3	<div>Se f é contínua em $\bar{p} \in D_f$ e \bar{U} são sucessões de pontos de D_f convergentes para \bar{t}, repectivamente, então $f(\bar{V})$</div> <div><table><tr><td rowspan="2">é uma sucessão de pontos de</td><td>$\mathbb{R}^{\bar{m}}$ c111</td><td rowspan="2">c11</td></tr><tr><td>$\mathbb{R}^{\bar{n}}$ c112</td></tr><tr><td>uma sucessão de números reais</td><td colspan="2">c12</td></tr></table></div> <div>que converge para $f(\bar{p})$.</div>	é uma sucessão de pontos de	$\mathbb{R}^{\bar{m}}$ c111	c11	$\mathbb{R}^{\bar{n}}$ c112	uma sucessão de números reais	c12		$c11 \wedge (c111 \vee (c112 \wedge (\bar{m} = \bar{n})))$
é uma sucessão de pontos de	$\mathbb{R}^{\bar{m}}$ c111		c11						
	$\mathbb{R}^{\bar{n}}$ c112								
uma sucessão de números reais	c12								
R4	<div>Se f é contínua em $\bar{p} \in D_f$ e $f(\bar{p}) = \bar{q}$</div> <div><table><tr><td>então</td><td>c11</td></tr><tr><td>nem sempre</td><td>c12</td></tr></table></div> <div>$\lim_{X \rightarrow \bar{p}} f_1 X = \bar{z}_1, \dots, \lim_{X \rightarrow \bar{p}} f_{\bar{m}} X = \bar{z}_{\bar{m}}$.</div>	então	c11	nem sempre	c12	c11			
então	c11								
nem sempre	c12								
R5	<div>Se $\bar{p} \in D_f$, $\lim_{X \rightarrow \bar{p}} f_1 X = \bar{z}_1, \dots, \lim_{X \rightarrow \bar{p}} f_{\bar{m}} X = \bar{z}_{\bar{m}}$ e</div> <div><table><tr><td>então</td><td>c11</td></tr><tr><td>nem sempre</td><td>c12</td></tr></table></div> <div>f é contínua em \bar{p}.</div>	então	c11	nem sempre	c12	c11			
então	c11								
nem sempre	c12								

Matemática

CICLO	TEMA-	OS	ND	CNT
5	1047			

Latex actualizado a

Latex imprimido a 14 de Fevereiro de 2007

Identificação do Modelo

Área	Matemática
Id do modelo	
Objectivo Secundário	2370
Informação adicional	Teorema de Green, aplicação prática (generalização).
Tipo de Modelo	4 - Texto com MathML alinhado à esquerda e SVG alinhado à direita, respostas com MathML
Ciclo de Ensino	5
Nível de Dificuldade	4

Objectivos das Respostas

R_1	Objmicro 5423	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_2	Objmicro 5570	Peso 1	
	Objmicro 5569	Peso 1	
	Objmicro	Peso	
R_3	Objmicro 5435	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_4	Objmicro 5436	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_5	Objmicro 5436	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	

R_6	Objmicro 5436	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_7	Objmicro 5436	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_8	Objmicro 5415	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	
R_9	Objmicro 5436	Peso 1	
	Objmicro	Peso	
	Objmicro	Peso	

Historial

Elaborado por	Manuel Martins	06 de Junho de 2005
Responsável	Manuel Martins	06 de Junho de 2005

Notas / Referências Bibliográficas

- Bortolossi, H.J., Cálculo Diferencial a Várias Variáveis, Uma Introdução à Teoria de Optimização, Ed. Loyola, 2002.
- Craizer, Marcos e Geovan Tavares Cálculo Integral a Várias Variáveis, Ed. Loyola, 2002.

MODELO

Observações/indicações de programação

Todos os cálculos com parâmetros devem ser simplificados ao máximo. Em $R4$ não deve aparecer o valor da fracção $\frac{1}{k}$, caso $k = 1$.

Notações e Abreviaturas

$o^n, \bullet^n \in \{+, -\}$; $o^n = -\bullet^n$.

MODELO Domínio dos Parâmetros

$x_0, y_0 \in \{0, 1, \dots, 5\}$; $a, b, c, d \in \{1, \sqrt{2}, \sqrt{3}, 2, 3, 4, 5\}$; $(c \leq a ; d \leq b ; (c \neq a \vee d \neq b))$; $k, i, j \in \{1, 2, 3, 4, 5\}$; $P_1 = xy^2$; $P_2 = -(k+1)x$; $P_3 = y^{k+1}$; $P_4 = -(k+1)y$; $P_5 = (k+1)x^k y$; $Q_1 = x^2 y$; $Q_2 = y^{k+1}$; $Q_3 = (k+1)y^k x$; $Q_4 = -(k+1)x$; $Q_5 = x^{k+1}$.

Texto

Sejam C_1 e C_2 as curvas de equação $\frac{(x \circ^1 x_0)^2}{c^2} + \frac{(y \circ^2 y_0)^2}{d^2} = 1$ e $\frac{(x \circ^1 x_0)^2}{a^2} + \frac{(y \circ^2 y_0)^2}{b^2} = 1$, com orientação positiva . D a região limitada pelas curvas C_1 e C_2 .

Respostas

Verdadeira se e só se:

R1	As funções:	$r_1(t) = (c \cos t \bullet^1 x_0, d \sin t \bullet^2 y_0) ; t \in [0; 2\pi].$ c11		$(c11 \wedge c21)$	
		$r_1(t) = (c \cos t \bullet^1 x_0, -d \sin t \bullet^2 y_0) ; t \in [0; 2\pi].$ c12			
	e	$r_2(t) = (a \cos t \bullet^1 x_0, b \sin t \bullet^2 y_0) ; t \in [0; 2\pi].$ c21			são
		$r_2(t) = (a \cos t \bullet^1 x_0, -b \sin t \bullet^2 y_0) ; t \in [0; 2\pi].$ c22			
	parametrizações regulares, para as curvas C_1 e C_2 , respectivamente, que respeitam a orientação.				
R2	A região D	é c11	conexa c21	$(c11 \wedge c21) \vee (c12 \wedge c22)$	
		não é c12	simplesmente conexa c22		
R3	Seja $F(x, y) = (P_i, Q_i)$. Então $\oint_{C_2} P_i \, dx + Q_i \, dy - \oint_{C_1} P_i \, dx + Q_i \, dy$			c11	
	$\begin{matrix} = & \text{c11} \\ \neq & \text{c12} \end{matrix}$				
	0.				

R4	A área da região D	é c11	o valor de c11
		não é c12	
	$\frac{1}{k} \oint_{C_2} xy^2 \, dx + (kx + x^2y) \, dy - \frac{1}{k} \oint_{C_1} xy^2 \, dx + (kx + x^2y) \, dy$ c21		
	$\frac{1}{k} \oint_{C_2} x^k \, dx + kx \, dy - \frac{1}{k} \oint_{C_1} x^k \, dx + kx \, dy$ c22		
	$\frac{1}{k} \oint_{C_2} -ky \, dx + y^k \, dy - \frac{1}{k} \oint_{C_1} -ky \, dx + y^k \, dy$ c23		

R5	<div> <div>Seja $F(x, y) = (xy^2, x)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c111</div> <div>$+\oint_{-C_1}$ c112</div> <div>$+\oint_{C_1}$ c113</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c114</div> <div>\neq c115</div> </div> <div>$\int \int_D 1 - 2xy \, dxdy.$</div> <div>c11</div>
	<div> <div>Seja $F(x, y) = (-kx, x)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c121</div> <div>$+\oint_{-C_1}$ c122</div> <div>$+\oint_{C_1}$ c123</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c124</div> <div>\neq c125</div> </div> <div>$\int \int_D 1 \, dxdy.$</div> <div>c12</div>
	<div> <div>Seja $F(x, y) = (-kx, x^2)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c131</div> <div>$+\oint_{-C_1}$ c132</div> <div>$+\oint_{C_1}$ c133</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c134</div> <div>\neq c135</div> </div> <div>$\int \int_D 2x \, dxdy.$</div> <div>c13</div>
	<div> <div>Seja $F(x, y) = (-ky, y^2)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c141</div> <div>$+\oint_{-C_1}$ c142</div> <div>$+\oint_{C_1}$ c143</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c144</div> <div>\neq c145</div> </div> <div>$\int \int_D k \, dxdy.$</div> <div>c14</div>
	<div> <div>Seja $F(x, y) = (ky^2, x)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c151</div> <div>$+\oint_{-C_1}$ c152</div> <div>$+\oint_{C_1}$ c153</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c154</div> <div>\neq c155</div> </div> <div>$\int \int_D 1 - 2ky \, dxdy.$</div> <div>c15</div>
	<div> <div>Seja $F(x, y) = (x^2y, x)$. Então, $\oint_{C_2} F(x, y) \, d\vec{r}$</div> <div> <div>$-\oint_{C_1}$ c161</div> <div>$+\oint_{-C_1}$ c162</div> <div>$+\oint_{C_1}$ c163</div> </div> <div>$F(x, y) \, d\vec{r}$</div> </div> <div> <div>$=$ c164</div> <div>\neq c165</div> </div> <div>$\int \int_D 1 - x^2 \, dxdy.$</div> <div>c16</div>

Validação de R5		$\left\{ \begin{array}{l} c11 \wedge \left\{ \begin{array}{l} c114 \wedge (c111 \vee c112) \\ \vee \\ c115 \wedge c113 \end{array} \right. \\ \vee \\ c12 \wedge \left\{ \begin{array}{l} c124 \wedge (c121 \vee c122) \\ \vee \\ c125 \wedge c123 \end{array} \right. \\ \vee \\ c13 \wedge \left\{ \begin{array}{l} c134 \wedge (c131 \vee c132 \vee (c133 \wedge x_0 = 0)) \\ \vee \\ c135 \wedge c133 \wedge (x_0 \neq 0) \end{array} \right. \\ \vee \\ c14 \wedge c144 \wedge (c141 \vee c142 \vee c143) \\ \vee \\ c15 \wedge \left\{ \begin{array}{l} c154 \wedge (c151 \vee c152) \\ \vee \\ c155 \wedge c153 \end{array} \right. \\ \vee \\ c16 \wedge \left\{ \begin{array}{l} c164 \wedge (c161 \vee c162 \vee (c163 \wedge c = 2 \wedge x_0 = 0)) \\ \vee \\ c165 \wedge c163 \wedge (c \neq 2 \vee x_0 \neq 0) \end{array} \right. \end{array} \right.$								
R6	<div>Seja $F(x, y) = \begin{array}{ c c } \hline (x^2, x) & c11 \\ \hline (-y, y^2) & c12 \\ \hline (x^2, 2x) & c13 \\ \hline (2y, 4x) & c14 \\ \hline \end{array}$.</div> <div><table><tr><td>O dobro da</td><td>c21</td></tr><tr><td>A</td><td>c22</td></tr></table> área da região D <table><tr><td>é</td><td>c31</td></tr><tr><td>não é</td><td>c32</td></tr></table></div> <div>a diferença entre o trabalho realizado pelo campo F ao longo da curva C_2 e o trabalho realizado pelo campo F ao longo da curva C_1.</div>	O dobro da	c21	A	c22	é	c31	não é	c32	$\left\{ \begin{array}{l} c31 \wedge \left\{ \begin{array}{l} c21 \wedge (c13 \vee c14) \\ \vee \\ c22 \wedge (c11 \vee c12) \end{array} \right. \\ \vee \\ c32 \wedge \left\{ \begin{array}{l} c21 \wedge (c11 \vee c12) \\ \vee \\ c22 \wedge (c13 \vee c14) \end{array} \right. \end{array} \right.$
O dobro da	c21									
A	c22									
é	c31									
não é	c32									

R7	Seja $F(x, y) =$	(x^2, x)	c11		c32						
		$(-y, y^2)$	c12								
		$(x^2, 2x)$	c13								
		$(2y, 4x)$	c14								
	<table><tr><td>O dobro da</td><td>c21</td></tr><tr><td>A</td><td>c22</td></tr></table>	O dobro da	c21	A	c22	área da região D	<table><tr><td>é</td><td>c31</td></tr><tr><td>não é</td><td>c32</td></tr></table>	é	c31	não é	c32
O dobro da	c21										
A	c22										
é	c31										
não é	c32										
balho realizado pelo campo F ao longo da curva C_2 com o trabalho realizado pelo campo F ao longo da curva C_1 .											

R8	O campo vectorial F definido por $F(x, y) =$	$(c11 \wedge i = j) \vee (c12 \wedge i \neq j)$				
	$(P_i(x, y), Q_j(x, y))$					
	<table><tr><td>é</td><td>c11</td></tr><tr><td>não é</td><td>c12</td></tr></table>	é	c11	não é	c12	
é	c11					
não é	c12					
	conservativo.					

R9	O volume do sólido $V = \{(x, y, z) \in \mathbb{R}^3 : (x, y) \in$	$(c11 \wedge (c21 \vee c22 \vee c24 \vee c26)) \vee$												
	$D \wedge 0 \leq z \leq kx^2\}$	$(c12 \wedge (c23 \vee c25))$												
	<table><tr><td>é</td><td>c11</td></tr><tr><td>nem sempre é</td><td>c12</td></tr></table>	é	c11	nem sempre é	c12									
é	c11													
nem sempre é	c12													
	dado													
	por $\oint_{C_2} F(x, y) \, d\vec{r} - \oint_{C_1} F(x, y) \, d\vec{r}$, onde $F(x, y) =$													
	<table><tr><td>$(-kx^2y, ky^2)$</td><td>c21</td></tr><tr><td>$(2xy - kx^2y, x^2)$</td><td>c22</td></tr><tr><td>$(2x, x^2 + \frac{kx^3}{3})$</td><td>c23</td></tr><tr><td>$(x^2, \frac{kx^3}{3})$</td><td>c24</td></tr><tr><td>$(\frac{x^2}{k}, \frac{(k+1)x^3}{3} + y^2)$</td><td>c25</td></tr><tr><td>$(x^2y, \frac{(k+1)x^3}{3})$</td><td>c26</td></tr></table>	$(-kx^2y, ky^2)$	c21	$(2xy - kx^2y, x^2)$	c22	$(2x, x^2 + \frac{kx^3}{3})$	c23	$(x^2, \frac{kx^3}{3})$	c24	$(\frac{x^2}{k}, \frac{(k+1)x^3}{3} + y^2)$	c25	$(x^2y, \frac{(k+1)x^3}{3})$	c26	
$(-kx^2y, ky^2)$	c21													
$(2xy - kx^2y, x^2)$	c22													
$(2x, x^2 + \frac{kx^3}{3})$	c23													
$(x^2, \frac{kx^3}{3})$	c24													
$(\frac{x^2}{k}, \frac{(k+1)x^3}{3} + y^2)$	c25													
$(x^2y, \frac{(k+1)x^3}{3})$	c26													